



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y TELECOMUNICACIÓN

Titulación:

INGENIERO TÉCNICO EN INFORMÁTICA DE GESTIÓN.

Título del proyecto:

MONITORIZACIÓN DE FINCAS DE CULTIVO MEDIANTE EL USO DE
SENSORES.

Alumno: Javier Ladrón Arroyo

Tutor: José Javier Astrain Escola

Pamplona, 29 de Junio de 2012

ÍNDICE

1. Introducción	
1.1 Descripción y objetivos.....	3
1.2 Estado del arte.....	4
1.3 Solución que propongo.....	10
2. Análisis y diseño	
2.1 Semántica del proyecto.....	14
2.2 Especificación de requisitos.....	20
2.3 Diagrama Entidad/Relación.....	24
2.4 Modelo Relacional (paso a tablas).....	25
2.5 Diagrama de Flujo de Datos.....	27
2.6 Diagrama de Casos de Uso.....	30
2.7 Diagrama de Clases.....	31
2.8 Diagrama de Colaboración.....	32
2.9 Pseudocódigo de los módulos de la página web.....	33
2.10 Arquitectura del sistema.....	38
2.11 Metodología de trabajo.....	39
2.12 Estimación temporal.....	40
3. Desarrollo de la propuesta	
3.1 Codificación de los módulos diseñados.....	42
3.2 Problemas que han surgido.....	86
3.3 Soluciones adoptadas.....	90
4. Pruebas	
4.1 Unitarias.....	107
4.2 De carga.....	119
5. Presupuesto	120
6. Conclusiones y líneas futura	122
7. Bibliografía	125
8. Anexo I. Manual de uso de la herramienta	126

1.1 DESCRIPCIÓN Y OBJETIVOS

El problema a resolver trata de implementar un mecanismo que tome, analice y muestre las principales características de las fincas y los invernaderos de la Universidad Pública de Navarra.

Se trata de que, de una manera automatizada los responsables de las fincas puedan llevar el control de todas las características posibles de su parcela. Con este proyecto se pretende tratar de evitar que el responsable, si quiere saber características de la finca y sus cultivos tales como la temperatura, el nivel CO2 ... tenga que ir con aparatos específicos y hacerlo él “a mano”.

De una manera más general, la mayor utilidad de este proyecto para el usuario sería explotar al máximo todo el beneficio que pudiese dar su parcela, debido a la constante toma de medidas y a los diferentes resultados obtenidos en la aplicación. Se trata de una explotación “inteligente”, en la que la toma de decisiones está apoyada por un sistema de medidas automatizado.

Para que el proyecto resulte rentable y sea útil, éste debe interactuar de manera totalmente ajena e independiente a la labor del usuario, es decir, que el usuario no tenga que preocuparse en absoluto por las tareas llevadas a cabo por el sistema. Al final de este proyecto el usuario únicamente debe poder consultar, comparar, analizar y decidir sobre todos los datos proporcionados por el sistema a través de las diferentes herramientas que ofrezca el propio proyecto.

El proyecto debe ser íntegro y sobre todo duradero, que de una solución al problema comentado de manera continuada, de manera que el usuario pueda disfrutar de él y aprovechar sus numerosas ventajas de manera continuada. Para ello se debe elegir bien la tecnología con la que se va a desarrollar.

Tan importante como la eficiencia del proyecto es la usabilidad del mismo. La interacción del sistema con el usuario debe poder ser completa, sencilla y amena. Que dé soluciones para todo lo que pueda abarcar el proyecto pero de una manera que el usuario pueda entender fácilmente, casi intuir.

Además el proyecto debe estar totalmente sincronizado. Es decir que si el sistema va tomando medidas de las características, prácticamente a la vez el usuario pueda visualizar en el momento dichas características en el sistema.

En resumen el proyecto debe tratar el problema de la monitorización de fincas para un usuario de manera que pueda sacar el máximo beneficio de su parcela únicamente consultando al sistema y que dicha consulta sea sencilla e intuitiva.

Redes de Sensores

Para el problema propuesto en el punto anterior, uno de los principales temas a abordar será el tema de la monitorización de fincas.

Para ello se utilizará una red de sensores. Se podría decir que una red de sensores es una red de pequeños dispositivos electrónicos, también llamados nodos, equipados con sensores, que colaboran en una tarea común.

Las redes de sensores están formadas por un grupo de sensores con ciertas capacidades sensitivas y de comunicación inalámbrica los cuales permiten formar redes sin una infraestructura física (cobre, fibra óptica ...).

Esta clase de redes se caracterizan por su facilidad de despliegue y por ser autoconfigurables, pudiendo convertirse en todo momento en emisor, receptor, ofrecer servicios de encaminamiento entre nodos sin visión directa, así como registrar datos referentes a los sensores locales de cada nodo.

Otra de sus características es su gestión eficiente de la energía, que les permite obtener una alta tasa de autonomía que las hacen plenamente operativas.

Las principales áreas de aplicación de las redes de sensores son:

- Eficiencia energética: Red de sensores se utilizan para controlar el uso eficaz de la electricidad, como el caso de Japón y España.
- Entornos de alta seguridad: Existen lugares que requieren altos niveles de seguridad para evitar ataques terroristas, tales como centrales nucleares, aeropuertos, edificios del gobierno de paso restringido. Aquí gracias a una red de sensores se pueden detectar situaciones que con una simple cámara sería imposible.
- Sensores ambientales: El control ambiental de vastas áreas de bosque o de océano, sería imposible sin las redes de sensores. El control de múltiples variables, como temperatura, humedad, fuego, actividad sísmica así como otras. También ayudan a expertos a diagnosticar o prevenir un problema o urgencia y además minimiza el impacto ambiental de la presencia humana.

- Sensores industriales: Dentro de fábricas existen complejos sistemas de control de calidad, el tamaño de estos sensores les permite estar allí donde se requiera.
- Automoción: Las redes de sensores son el complemento ideal a las cámaras de tráfico, ya que pueden informar de la situación del tráfico en ángulos muertos que no cubren las cámaras y también pueden informar a conductores de la situación, en caso de atasco o accidente, con lo que estos tienen capacidad de reacción para tomar rutas alternativas.
- Medicina: Es otro campo bastante prometedor. Con la reducción de tamaño que están sufriendo los nodos sensores, la calidad de vida de pacientes que tengan que tener controlada sus constantes vitales (pulsaciones, presión, nivel de azúcar en sangre, etc), podrá mejorar sustancialmente.
- Domótica: Su tamaño, economía y velocidad de despliegue, lo hacen una tecnología ideal para domotizar el hogar a un precio asequible.

En cuanto a las características, las principales son:

- Topología Dinámica: En una red de sensores, la topología siempre es cambiante y éstos tienen que adaptarse para poder comunicar nuevos datos adquiridos.
- Variabilidad del canal: El canal radio es un canal muy variable en el que existen una serie de fenómenos como pueden ser la atenuación, desvanecimientos rápidos, desvanecimientos lentos e interferencias que puede producir errores en los datos.
- No se utiliza infraestructura de red: Una red sensora no tiene necesidad alguna de infraestructura para poder operar, ya que sus nodos pueden actuar de emisores, receptores o enrutadores de la información.
Sin embargo, hay que destacar en el concepto de red sensora la figura del nodo recolector (también denominados sink node), que es el nodo que recolecta la información y por el cual se recoge la información generada normalmente en tiempo discreto.
Esta información generalmente es adquirida por un ordenador conectado a este nodo y es sobre el ordenador que recae la posibilidad de transmitir los datos por tecnologías inalámbricas o cableadas según sea el caso.

- Tolerancia a errores: Un dispositivo sensor dentro de una red sensora tiene que ser capaz de seguir funcionando a pesar de tener errores en el sistema propio.
- Comunicaciones multisalto o broadcast: En aplicaciones sensoras siempre es característico el uso de algún protocolo que permita comunicaciones multi-hop, aunque también es muy común utilizar mensajería basada en broadcast.
- Consumo energético: Es uno de los factores más sensibles debido a que tienen que conjugar autonomía con capacidad de proceso, ya que actualmente cuentan con una unidad de energía limitada. Un nodo sensor tiene que contar con un procesador de consumo ultra bajo así como de un transceptor radio con la misma característica, a esto hay que agregar un software que también conjugue esta característica haciendo el consumo aún más restrictivo.
- Limitaciones hardware: Para poder conseguir un consumo ajustado, se hace indispensable que el hardware sea lo más sencillo posible, así como su transceptor radio, esto nos deja una capacidad de proceso limitada.
- Costes de producción: Dada que la naturaleza de una red de sensores tiene que ser en número muy elevada, para poder obtener datos con fiabilidad, los nodos sensores una vez definida su aplicación, son económicos de hacer si son fabricados en grandes cantidades.

Tecnología ZigBee

En este apartado se introducirá un poquito la tecnología inalámbrica ZigBee ya que es la que se ha usado para este proyecto. Se dará una breve explicación en qué consiste dicha tecnología, sus ventajas, inconvenientes, aplicaciones en las que se utiliza principalmente y alguna comparación con otras tecnologías. Más adelante, en el apartado de desarrollo del proyecto se hablará con más detalle de esta tecnología.

De todas las tecnologías inalámbricas utilizadas en una red de sensores se ha optado por el desarrollo de ZigBee. El motivo de viene dado por las numerosas ventajas que más tarde explicaré en este apartado.

ZigBee es un estándar de comunicaciones inalámbricas diseñado por la ZigBee Alliance. Es un conjunto estandarizado de soluciones que pueden ser implementadas por cualquier fabricante.

ZigBee está basado en el estándar IEEE 802.15.4 de redes inalámbricas de área personal y tiene como objetivo las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

Comunica una serie de dispositivos haciendo que trabajen más eficiente entre sí. Es un transmisor y un receptor que usa baja potencia para trabajar y tiene como objetivo las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

Es ideal para conexiones con diversos tipos de topología, lo que a su vez lo hace más seguro, barato y que no haya ninguna dificultad a la hora de su construcción porque es muy sencilla.

Existen tres tipos de dispositivos. Coordinador, Router, nodo final. Todo esto y mucho más se explicará más detalladamente en el apartado de implementación.

Las principales ventajas de usar la tecnología ZigBee son:

- Ideal para conexiones punto a punto y punto a multipunto. Con ello favorece una mayor diversidad de comunicación.
- Diseñado para el direccionamiento de información y el refrescamiento de la red.
- Opera en la banda libre de ISM 2.4 GHz para conexiones inalámbricas.

- Óptimo para redes de baja tasa de transferencia de datos.
- Alojamiento de 16 bits a 64 bits de dirección extendida.
- Reduce tiempos de espera en el envío y recepción de paquetes.
- Detección de Energía (ED).
- Baja ciclo de trabajo - Proporciona larga duración de la batería.
- Soporte para múltiples topologías de red: Estática, dinámica, estrella y malla.
- Hasta 65.000 nodos en una red.
- 128-bit AES de cifrado. Provee conexiones seguras entre dispositivos.
- Son más baratos y de construcción más sencilla.

Las principales desventajas son:

- La tasa de transferencia es muy baja.
- Solo manipula textos pequeños comparados con otras tecnologías.
- Zigbee trabaja de manera que no puede ser compatible con bluetooth en todos sus aspectos porque no llegan a tener las mismas tasas de transferencia, ni la misma capacidad de soporte para nodos.
- Tiene menor cobertura porque pertenece a redes inalámbricas de tipo WPAN.

Observación: ZigBee está basado en un protocolo que tiene una estructura jerárquica. Los nodos finales están controlados por un nodo router y estos a su vez por el nodo coordinador.

En caso de ausencia de nodos router el coordinador (tiene que haber uno obligado) será el que recoja todos los datos obtenidos por el resto de nodos fin.

En caso de que el coordinador fallase, toda la red se iría al traste. Es importante comprender eso.

Los principales campos donde se puede utilizar la tecnología ZigBee podrían ser los siguientes:

- Automatización de edificios. Temas de seguridad. Control de temperatura, de luces. Control de acceso.
- Salud. Monitorización de pacientes, control del estado físico.
- Control industrial. Manejo de cargas. Control de procesos, de ambiente, de energía ...
- Control en la agricultura, el clima, el tiempo ...
- Electrodomésticos. Televisión, DVD, controles remotos ...
- Ordenadores personales y periféricos. Temas del ratón, teclados, joystick ...
- Control de hogares y pequeñas oficinas. Temas de seguridad. Control de temperatura, de luces. Control de acceso. Irrigación de jardines.

Como último punto de este apartado sobre la tecnología ZigBee se va a comparar dicha tecnología con otras tecnologías de tipo inalámbrico.

Para no profundizar mucho (más adelante se hará) se puede decir que por ejemplo Bluetooth, WiFi operan en una frecuencia de 2,4 GHz. ZigBee puede operar también a 2,4 GHz aunque también puede hacerlo a frecuencias más bajas (868 y 915 MHz).

La tasa de transferencia de WiFi es de hasta 11 Mbps, de Bluetooth 1Mbps. De ZigBee en cambio es bastante más baja. Puede ser de 256 kbps, 40 kbps, 20 kbps según la frecuencia en la que se haya configurado la red.

El número máximo de dispositivos que podemos colocar en una misma red WiFi es de 32, en una red Bluetooth es 8, mientras que en una red ZigBee, en teoría se podría colocar hasta un máximo de 65535.

Por último, como requisitos de alimentación un dispositivo WiFi podría alcanzar horas de batería usándose sin parar. Un dispositivo de bluetooth podría estar días aguantado con una sola batería. En cambio un dispositivo que use ZigBee podría estar hasta 5 años funcionando sin tener que cambiar la batería. [1]

1.2 SOLUCIÓN QUE PROPONGO

Como solución a la descripción del problema planteado he decidido monitorizar las fincas con una red de dispositivos Wasmote. A la vez he optado por usar la tecnología inalámbrica ZigBee para la comunicación entre los dispositivos.

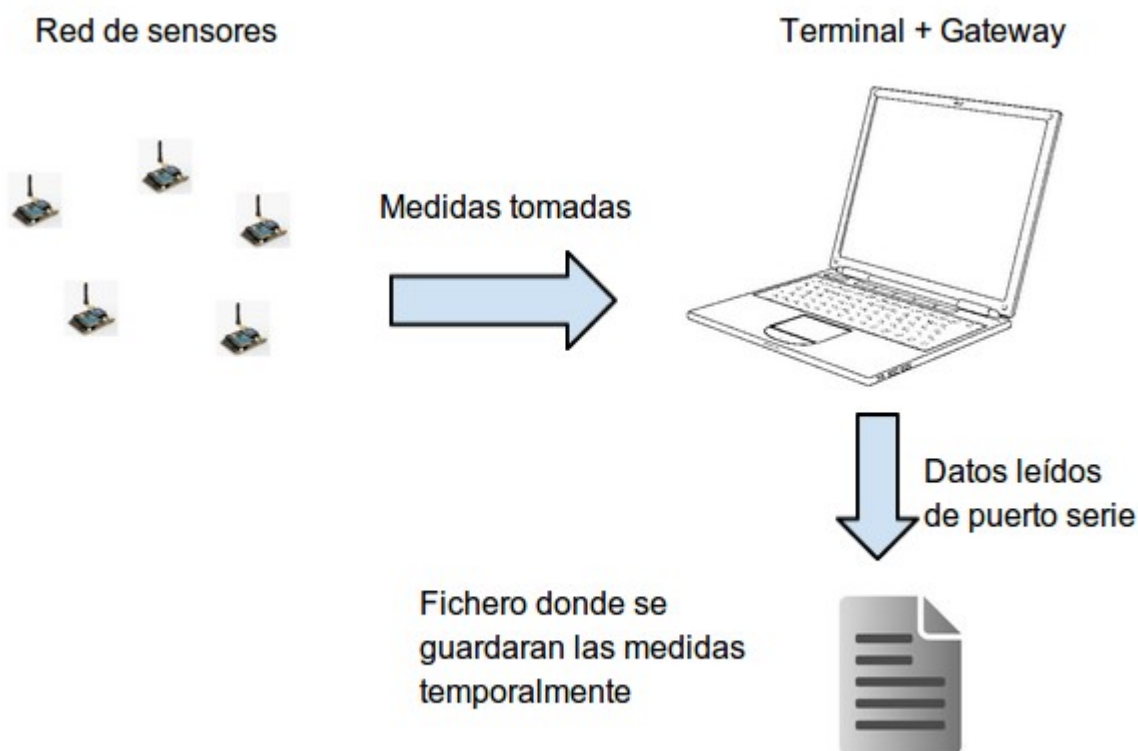
Dichos dispositivos son programados en el lenguaje C++. Todos los dispositivos están configurados de manera que son nodos finales. Cada uno de ellos, según el programa que estén ejecutando, miden con un determinado sensor una característica y la envían mediante el protocolo ZigBee los datos al coordinador.

Dichos datos es un son una concatenación de cadenas de caracteres formados por el tipo de sensor, el valor de la medida tomada, el identificador del sensor (sensor definido para un usuario), la fecha y la hora. Todo ello va precedido por una serie de caracteres que forman la cabecera del protocolo ZigBee.

El coordinador (que será un gateway) está conectado a un puerto USB simulando un puerto serie (más sencillo) en un terminal (ordenador).

En dicho terminal habrá un programa en ejecución que leerá todo lo que le llegue por el puerto serie al que está conectado el gateway y lo escribirá en un fichero.

Para un sistema operativo Windows dicho programa se llama HiperTerminal mientras que si el sistema operativo es Linux (distribución libre de Unix) el programa se llama Minicom.



A continuación voy a explicar cómo guardo en una base de datos las medidas tomadas por la red de sensores escritas en un fichero de texto.

Se trata de un programa hecho en java, que a grandes rasgos, monitoriza el fichero con las medidas tomadas hasta ese momento y las guarda en una base de datos.

La interfaz del programa se muestra en la Figura 1:

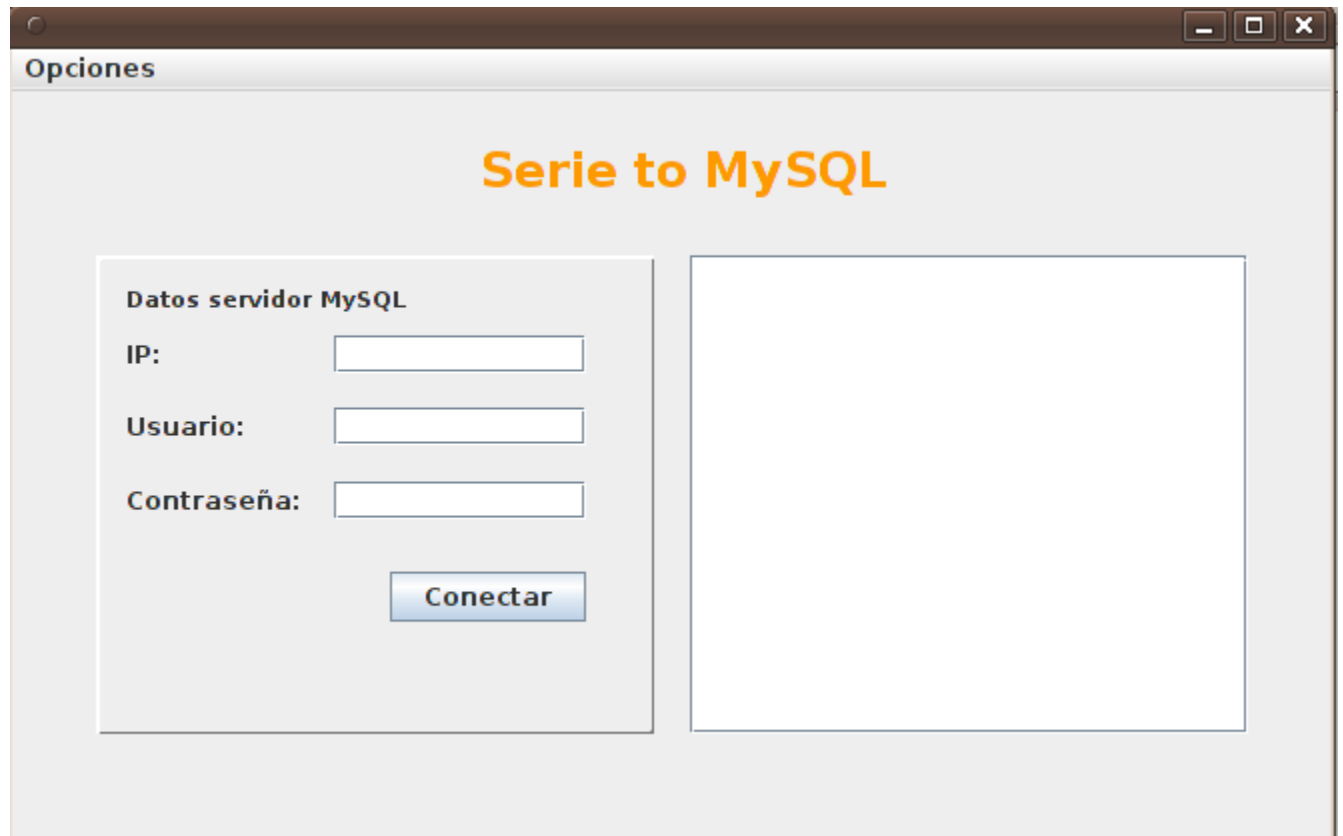


Figura 1. Interfaz del programa que monitoriza el fichero con las medidas

Se debe dar el fichero a monitorizar mediante la opción del menú Opciones. A continuación se debe dar la IP del servidor de base de datos junto con su login y password.

Profundizando un poco más en cómo funciona el programa se puede decir que el programa, primero se conecta al servidor de la base de datos con los datos proporcionados por el usuario (IP, login, password).

En segundo lugar lee medidas una a una. Las trocea, las prepara y comprueba mediante un consulta SQL si ya ha insertado dicha medida. Si ya la ha insertado

previamente lee la siguiente medida y repite el proceso. Si no estaba previamente introducida la inserta en la base de datos y pasa a la siguiente medida.

Así continúa hasta que llega que llega a fin de fichero monitorizado por el programa. Una vez que acaba cierra la conexión con la base de datos y se queda dormido hasta que pase el tiempo de reposo para volver a repetir todo el proceso.

La última parte para la solución que propongo es la construcción de una página web privada de manera que el usuario pueda consultar, comparar ... las medidas tomadas por los sensores y guardadas en la base dados.

Además, quiero dotar de otras funcionalidades a la página de manera que sea una herramienta útil para el usuario.

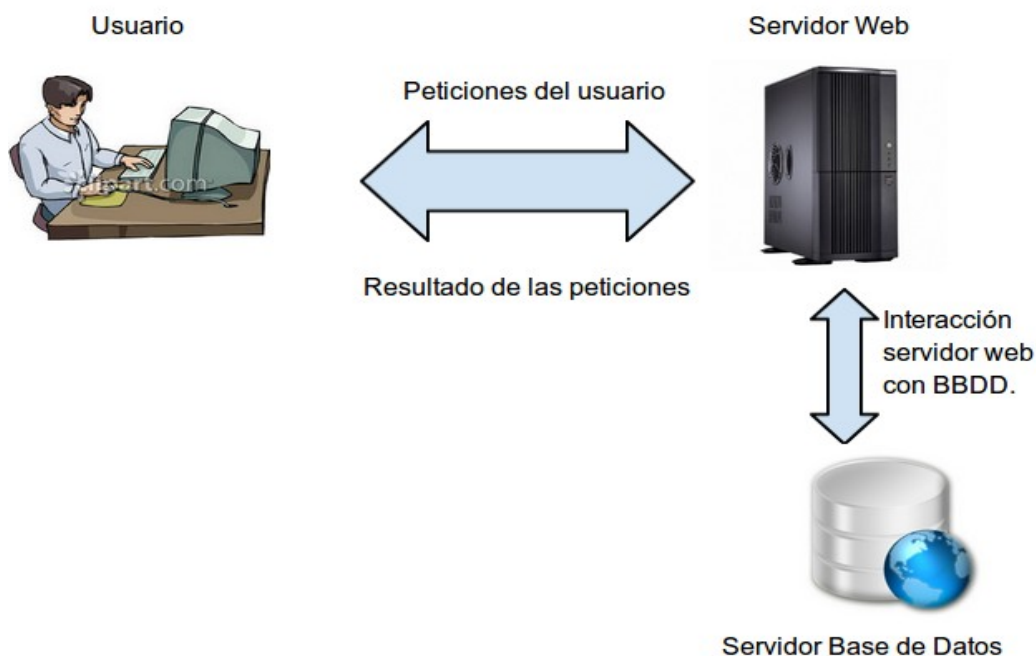
Una de esas funcionalidades podría ser la creación de alarmas, para establecer límites para las medidas que pudiesen ser resultar críticas.

Otra sería un apartado de gráficas para que el usuario pueda ver de manera gráfica los datos obtenidos por los sensores. Tanto el apartado de visualización de las medidas tomadas como para el de gráficas habrá diferentes opciones en lo referente al tiempo (por años, o meses o días).

Interesaría también tener un sitio donde la web ofreciera al usuario la posibilidad de guardar y descargar los ficheros que él quisiera.

Por último se me ocurre que sería de utilidad tener un mapita que represente la finca del usuario donde de manera sencilla, por ejemplo con un click de ratón el usuario pudiese poner y quitar dispositivos del mapita.

Otra cosa que supongo que sería útil sería el tener una especie de perfil donde cada usuario tuviese sus datos personales, junto a una foto y con todas las alarmas creadas.



Esta es la mejor propuesta que se me ocurre para resolver la propuesta del problema planteado por la descripción y los objetivos del proyecto.

Me he decantado por usar una red de sensores, con dispositivos Waspote, de Libelium ya que son una clase de dispositivos que soportan la tecnología inalámbrica ZigBee, que es bastante eficiente.

Además dichos dispositivos son programados en un lenguaje de alto nivel orientado a objetos como C++, lo cual hace una programación menos tediosa y más amena.

La otra gran decisión del proyecto es saber que herramienta final es la que va a usar el usuario. Se podría haber hecho perfectamente un programa en local (por ejemplo en java) perfectamente. Pero finalmente me decidí por hacerlo en web para el usuario pudiese acceder al sistema desde cualquier terminal (ordenador, smartphone, tableta) con acceso a Internet del mundo.

Eso proporciona al usuario una flexibilidad mucho mayor al usuario que si la aplicación fuese un programa en local.

Por último cabe destacar como apunte, que el hecho de dotar a la web de herramientas como el alojamiento de ficheros ... es proporcionar al usuario la mayor unidad posible dentro de su trabajo, es decir, que tenga la mayor parte de sus servicios en una misma web.

2.1 SEMÁNTICA DEL PROYECTO

Aquí voy a mostrar más detalladamente una especificación del proyecto.

Se pretende construir un sistema informático en el cual se monitoricen las fincas y los invernaderos de la universidad mediante el uso de sensores de manera que sus responsables (usuarios) puedan ver, analizar y comparar los resultados y aplicar acciones sobre ellos en una página web privada.

Para ello cada usuario (usuario=responsable finca) dispondrá de unos dispositivos (también llamado nodo) capaces de tomar medidas sobre las principales características útiles de una finca mediante unos sensores incorporados en ellos. Cada usuario gestiona mínimo un dispositivo o varios. Cada dispositivo pertenece a un único usuario. Para cada una de las redes (parcelas de los usuario monitorizadas con dispositivos) de cada usuario los dispositivos que hay en ellas envían los datos a un gateway mediante el uso de la tecnología ZigBee (Esta tecnología se explicará posteriormente). Una vez recogidos los datos de los dispositivos por el gateway, mediante la ayuda de un par de programas se introducirán en un servidor de base de datos.

Obviamente, la página web a la que tienen acceso los usuarios, tiene acceso a esa misma base de datos.

De cada usuario se quiere saber su nombre, apellidos, un nombre de usuario (login), una contraseña, el tipo de usuario (normal o administrador), una foto del usuario (path donde se alojará la imagen), el correo electrónico y el número de de teléfono. Como clave primaria se asignará a cada usuario un identificador de usuario.

Ninguno de los datos del usuario puede ser nulo.

Cada dispositivo puede llevar incorporados uno o varios sensores. Cada sensor es único de ese dispositivo. Con esto quiero decir que varios sensores pueden medir temperatura por ejemplo, pero cada uno de ellos pertenecerán siempre a ese dispositivo.

De cada dispositivo se pretende saber un identificador único que actuará como clave primaria, además de su ip y su dirección mac. Todos los datos son obligatorios

Cada uno de los sensores de cada dispositivo puede tomar una o varias medidas. Esas medidas son tomadas, obviamente por un único sensor.

De cada medida se pretende saber un identificador único que será su clave primaria, el tipo (que sensor ha capturado la medida), el valor, la fecha y la hora a la que se tomo la medida.

Ninguno de los atributos de medida debe ser nulo.

De cada sensor (también lo llamaré componente) es interesante saber el nombre del sensor y un identificador único que actuará como clave primaria (ambos valores no deben ser nulos).

Una de las principales utilidades que tendrá la web será que el usuario podrá generar alarmas para ver las medidas críticas que tomen los sensores, y poder tomar medidas. Dicho esto se debe considerar que un usuario podrá generar o no varias alarmas. Cada alarma es única para un usuario.

De cada alarma se desea conocer el nombre de la alarma, que será único para cada usuario, es decir, dos alarmas no podrán llamarse igual para un mismo usuario pero si podran llamarse de igual manera si pertenecen a usuarios distintos. Además del nombre se quiere conocer de cada alarma su tipo (sobre que sensor se crea dicha alarma), su signo (mayor, menor o igual) junto con el umbral (valor de la alarma). Además se requerirá una descripción (motivo por el cual se crea la alarma) y una acción correctora, donde el usuario propondrá las medidas necesarias para que las condiciones críticas tomadas por los sensores mejoran. Todos los datos de la alarma no deben ser nulos.

Todo dispositivo está situado en una ubicación. De cada ubicación se pretende saber un identificador que será la clave primaria. Además se quiere saber la localidad, zona (lugar donde esta situado dentro de la ciudad), la latitud, la longitud y la altitud. Un nodo pertenece a una y solo a una Ubicación. En una ubicación habrá mínimo un nodo. También puede haber varios. Todos los datos de Ubicación deben ser obligatorios. No nulos.

A su vez cada nodo pertenece a una red, que puede coincidir con la finca/parcela del usuario o tener un sentido más general (varias fincas por ejemplo). Un nodo pertenece a una única red (pero pertenece a una). En una red puede haber uno o varios nodos. De cada red se quiere saber su nombre, una breve descripción de la misma, y su estructura, que es para saber dónde tiene cada usuario sus dispositivos. Una herramienta muy útil para saber donde puede colocarlos mejor. Como clave primaria se debe dar un identificador que será único para cada red. No debe haber ningún dato nulo.

Nota: Tanto para Ubicación como para red tiene que quedar claro que cada nodo pertenece a una y sólo a una red y una ubicación. Pero dicha relación puede ser en

ambos casos actualizada, es decir que un nodo, puede pertenecer a una red, pero luego dejar de pertenecer a ella y pasar a pertenecer a otra. Lo mismo para una ubicación. Pero nunca podrá estar en dos redes o dos ubicaciones a la vez.

Por último también se quiere disponer de una entidad llamada fichero donde se quiere saber el nombre del fichero, el tipo de fichero y el tamaño del fichero. Además debe poseer un identificador único para cada fichero que será su clave primaria.

Con todo esto un usuario puede ser dueño de varios ficheros o ninguno. Todo fichero pertenece a un usuario. Todos los atributos de fichero deben ser no nulos.

La página web debe ser una herramienta sencilla e intuitiva para el usuario, que no sea demasiado complicada.

Por supuesto en la web el usuario debe poder visualizar las medidas tomadas por los sensores.

Debe proporcionar al usuario diferentes opciones de consulta.

Debe poder consultar por años, por meses y por días.

Para cada una de estas los resultados (cada medida en una línea en la que aparecerán el número de medida, tipo, valor, fecha, hora y el nombre del dispositivo que la ha medido) aparecerán ordenados por el tipo de sensor y dentro de cada tipo de sensor por fecha y hora.

Para cada tipo (sensor) de medidas se debe dar la máxima, la mínima y la media.

Las medidas que no entren en conflicto con las alarmas (se explican en el siguiente apartado) creadas por el usuario deben aparecer en color verde. Las que se vean afectadas deberán aparecer en rojo junto con el nombre de la alarma con la que se ve afectada. El nombre de la alarma deberá ser un enlace que al pinchar sobre el recargará la página como estaba y mostrará en la parte inferior de la página las características de la alarma.

El sistema también deberá proporcionar al usuario la posibilidad de mostrar los resultados de la consulta a través de un fichero Word que se descargará automáticamente. Dicha opción será activada pulsado un checkbox.

Otro de los servicios principales que la página web es un apartado de alarmas, donde el usuario podrá crear nuevas alarmas o modificar y borrar alarmas ya existentes.

Para la creación de una nueva alarma la web le debe pedir al usuario, el nombre de alarma, el tipo, el signo el umbral, una breve descripción de la misma y una posible acción correctora. La web debe comprobar dos cosas. La primera de ellas es asegurarse que el usuario rellena todos y cada uno de los campos del formulario ya

descritos, y la segunda es asegurarse que para un mismo usuario el nombre de la alarma debe ser único.

Si todos los parámetros son correctos se mostrará un cuadro diciendo que la alarma ha sido creada satisfactoriamente y le redirigirá a la zona del muro, que más adelante explicaré. En caso de que ocurriese alguno de los dos fallos anteriormente descritos el sistema le avisará mediante otro recuadro que infracción ha cometido y le volverá a mostrar el formulario de nuevo.

Para la opción de modificación de alarmas en una primera instancia se debe dar un listado de las alarmas creadas por el usuario en el que se señalará el nombre de la alarma y la descripción de la misma junto con un icono que represente el botón de modificar. Si el usuario hace click en alguno de los iconos de cualquier alarma el sistema lanzará el formulario de creación de alarma pero con los campos rellenos con la características de esa alarma. Una vez que el usuario modifique los campos que considere oportunos el sistema actualizará dicha alarma.

Para la opción de borrado el sistema en un principio hará lo mismo que el de modificación pero mostrará las alarmas con sus descripciones creadas pero esta vez junto a un icono de borrado en lugar de modificación. Una vez que el usuario haga click en el icono de borrar de cualquiera de las alarmas que el usuario ha creado el sistema borrará automáticamente dicha alarma.

Es de perogrullo decir que todos los cambios que se producen en el apartado de alarmas influyen directamente en el apartado de historial ya explicado (las medidas afectadas por las alarmas creadas).

El siguiente a punto a tener en cuenta, es que el sistema debe tener una especie de perfil (propio de cada usuario) llamado Muro donde básicamente se mostrará los datos personales del usuario, la foto, nombre, apellidos, correo y teléfono.

A continuación se mostrará todas las alarmas creadas por el usuario junto con sus características.

El sistema debe incorporar un sistema de gráficas. En él, el usuario, al igual que en el apartado de historial el usuario podrá realizar consultas gráficas tanto por años, como por meses o días. Si por ejemplo la consulta es por meses o años el sistema debe dar desde el primer mes (o año) hasta el último mes (o año) la máxima la mínima y la media de cada uno de los tipos de medidas (sensores) y un gráfico en el que se visualizarán las medidas medias del intervalo escogido por el usuario.

En caso de que la consulta del usuario sea por días los días del intervalo que sean nulos (en ese día no hay medidas) no deben aparecer en la gráfica para garantizar la integridad de la misma.

El sistema debe proporcionar al usuario la posibilidad de elegir sobre tres tipos de gráfico. Gráfico de líneas, gráfico de barras y gráfico de sectores.

Dicha opción la elegirá el usuario mediante un grupo de radiobuttons que estará junto al intervalo elegido por el usuario.

Cabe recordar que los resultados deben ser que, para cada sensor (CO2, humedad ...) debe aparecer las máximas, mínimas y medias claramente diferenciadas de cada división del intervalo elegido por el usuario y a continuación una imagen con la grafica representando las medias de las divisiones del intervalo elegido por el usuario para ese sensor.

La página web también debe disponer de un apartado en el que el usuario pueda simular dónde tiene puestos los dispositivos en la finca.

Aparecerá un mapita representando la finca, dividido en cuadritos pequeños. Si el usuario hace click en cualquiera de ellos y está vacío a continuación se refrescará dicho mapa con la imagen de un dispositivo en el cuadrito seleccionado. La herramienta debe funcionar también al contrario, es decir, si el usuario desea cambiar de lugar el dispositivo o simplemente quitarlo al pinchar sobre la imagen deberá desaparecer dicha imagen del dispositivo. Deberá ser muy sencillo e intuitivo.

Otra de las funciones que deberá dar la pagina web es ofrecer al usuario un sistema de almacenamiento de ficheros, llamado Mis documentos, donde el usuario podrá subir en principio todos los ficheros que desee. A su vez el usuario en cualquier momento podrá tanto descargar ese fichero, como borrarlo del servidor web. Tanto para descargar, como para borrar aparecerán dos iconos al lado del fichero subido.

De esta manera se garantiza al usuario que la web sea una especie de centro estadístico donde además de poder llevar el control de la finca con todos las herramientas que acaban de ser explicadas también pueda tener a la vez un sitio donde alojar sus documentos. La única restricción en principio, es que los ficheros no deben tener un tamaño superior a 70000 bytes.

Todas estas herramientas descritas anteriormente son propias y personales de cada usuario. Así mismo también se desea tener un mecanismo de comunicación con los demás usuarios de la web en la propia web.

Para la página web deberá tener un chat compartido donde los usuarios podrán dejar mensajes al resto de usuarios.

Por último se desea tener un panel de configuración donde cada usuario podrá cambiar si lo desea su datos personales. Al hacer click en esa sección todos los campos del usuario (nombre, apellidos, foto, nombre usuario, password, correo y teléfono) deberán

aparecer rellenos con sus datos. Si el usuario una vez acabado de corregir los datos guarda los cambios correctamente (no hay campos nulos ...) se le redireccionará a su muro donde podrá comprobar que efectivamente sus datos han sido correctamente modificados.

Para unir las dos grandes partes del proyecto comentadas (red de sensores y página web) se ha de hacer un programa auxiliar en un lenguaje orientado a objetos (java por ejemplo) que monitorice el fichero, donde el lector de puerto serie (minicom en linux, hiperTerminal en Windows) al que está conectado el gateway está guardando las medidas recibidas de los sensores, cada cierto tiempo e inserte en la base de datos las medidas que previamente no hayan sido ya insertadas.

Para ello dicho programa extraerá del usuario la IP, el login y la password, junto con el fichero a monitorizar.

Una vez obtenidos dichos datos, el programa lanzará un hilo concurrente, en el que con la ayuda de clases (API's de operaciones) sobre ficheros (apertura, lectura ...) y sobre la base de datos (conexión, consultas, inserciones, cierre ...) podrá, cada cierto tiempo:

- Establecer una conexión con la base de datos.
- Mientras no sea fin de fichero, ir leyendo las medidas escritas en el fichero a monitorizar, comprobar mediante una consulta SQL si esa medida no ha sido previamente insertada en la base de datos, y si es así, insertarla en ella.
- Una vez llegado a fin de fichero (donde están las medidas tomadas por los sensores) el programa cerrará correctamente la conexión con la base datos asegurándose de no causar ninguna excepción.
- El programa mostrará por una pantalla (incluida en la interfaz principal) si se ha conectado correctamente o no a la base datos, si para cada medida insertada en la base de datos ha sido insertada correctamente o no y si se ha cerrado correctamente o no la conexión con la base de datos.

2.2 ESPECIFICACIÓN DE REQUISITOS

En esta parte del análisis voy a centrarme en la especificación de requisitos.

Requisitos funcionales:

Como básicamente se podría decir que el proyecto consta de tres partes, red de sensores, página web con base de datos y el software que une la red de sensores con la base de datos, los requisitos funcionales los voy a separar para cada una de las tres partes.

1. Requisitos funcionales para la red de sensores.

- Los dispositivos deben poder medir cualquiera de los sensores que tienen incorporados.
- Una vez hecha cada media, sea del tipo que sea, enviarán los datos a través de la red ZigBee al coordinador (en mi caso, el gateway).

2. Requisitos funcionales para el software auxiliar de inserción de medidas en la base de datos

- El programa debe poder proporcionar al usuario de una manera sencilla la elección del fichero a monitorizar.
- Obviamente tiene que pedir al usuario la IP, el login y el password del servidor de base de datos.
- El programa debe poder realizar cada cierto tiempo las inserciones escritas en fichero que no hayan sido ya insertadas en la base de datos.
- El programa mostrará en una ventana si la conexión con la base de datos ha sido correcta, si las medidas que han sido insertadas han sido correctamente insertadas y si se ha cerrado la conexión con la base de datos de forma correcta.

3. Requisitos funcionales para la página web.

- Se debe poder visualizar las medidas de manera que el usuario pueda elegir entre diferentes intervalos de tiempo (años, meses, días). Además se dará, la máxima, la mínima y la media de cada sensor.
- Se debe poder crear, modificar y borrar alarmas para el control de medidas que el usuario considere críticas.
- Una vez realizada una consulta, las medidas que se vean afectadas por las alarmas creadas por el usuario se deben diferenciar claramente del resto de las medidas que sean correctas además de aparecer el nombre de la alarma con la que se ve afectada.
- Se deberá disponer de un sistema de gráficas donde el usuario pueda visualizar de una manera gráfica medias de las medidas según un intervalo elegido por el usuario (años, meses, días). Los gráficos podrán ser de líneas, barras y sectores.
- Se deberá tener un perfil privado llamado muro donde aparezcan los datos personales del usuario, junto a una foto y las alarmas creada por el usuario.
- La web ofrecerá alojamiento al usuario para que pueda subir todos sus los ficheros que éste desee.
- También tendrá un mapita, que represente la finca del usuario, donde el usuario podrá, haciendo click en una zona concreta pueda poner y quitar dispositivos, para representar donde los tiene colocados en la realidad.
- Se implementará un chat compartido donde los usuarios podrán comunicarse entre todos ellos.

Requisitos no funcionales:

Ahora mostraré los requisitos no funcionales, y al igual que con los funcionales, los mostraré para cada una de las tres partes diferenciadas del proyecto.

1. Requisitos no funcionales para la Red de Sensores.

- Los dispositivos Waspote de Libelium son programados con un lenguaje orientado a objetos, C++. Se usa el API de libelium para crear las clases que controlan los sensores y la tecnología ZigBee.
- Se utilizará el IDE (editor de texto) proporcionado por Libelium, ya que facilita la compilación y la carga del código C++ en los dispositivos.

2. Requisitos no funcionales para el software auxiliar de inserción de medidas en la base de datos.

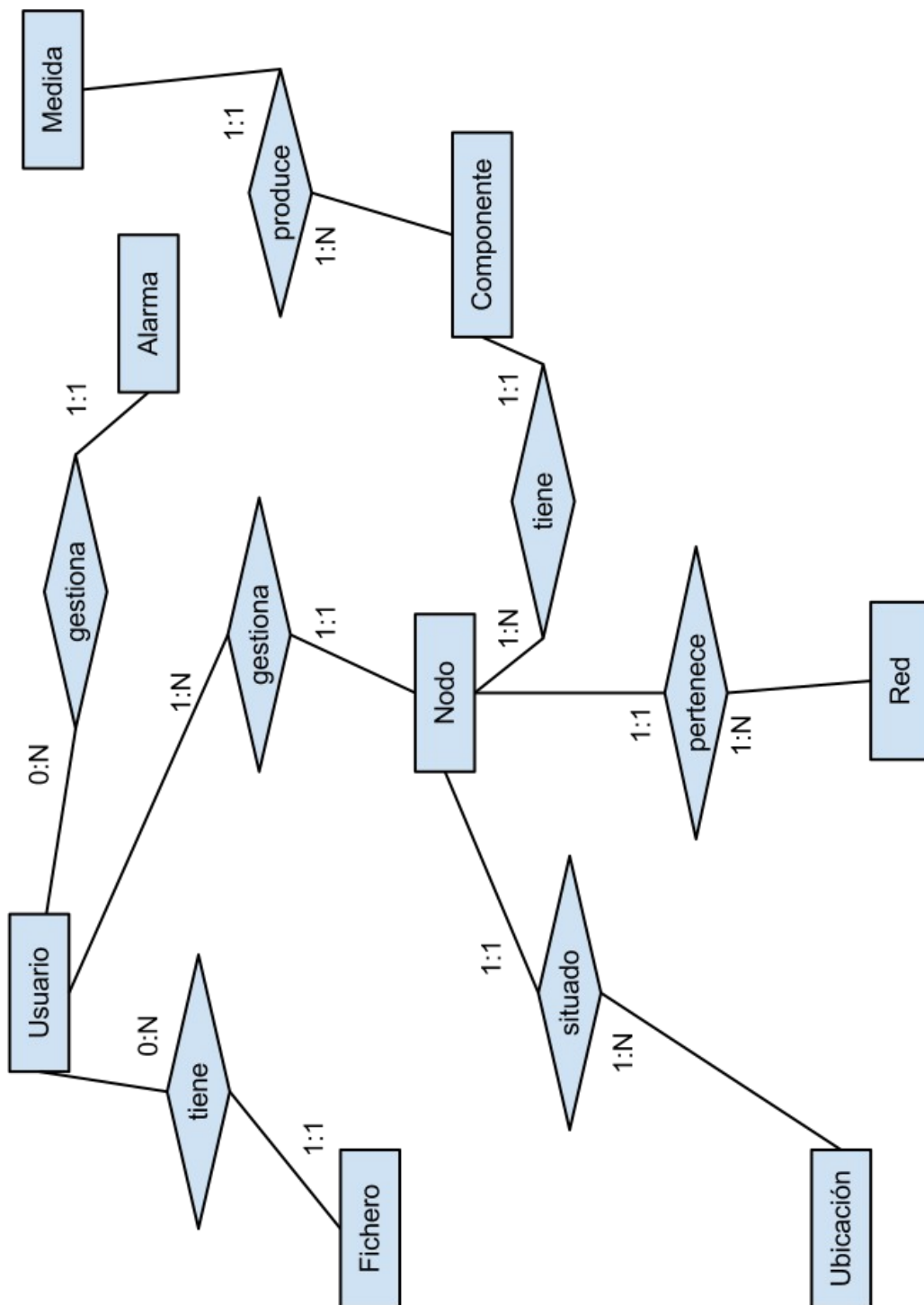
- Después de cada turno de inserciones el programa deberá asegurarse de cerrar la conexión con la base de datos, para asegurarse que cuando vuelva a monitorizar el fichero con las medidas vuelva a hacer la conexión con la base de datos de manera correcta.
- Debe estar implementado en un lenguaje orientado a objetos, preferiblemente en java, para facilitar la programación, y la interfaz gráfica para el usuario.
- El programa no debe bloquearse, es decir, se debe evitar que haya un único proceso que esté ejecutando la CPU y pueda dar algún tipo de bloqueo. Para ello se recomienda el uso de algún proceso concurrente para que se encargue de la apertura, lectura y cierre de fichero, de la apertura, inserción y cierre de la base de datos, que pueden ser las acciones que conlleven algún tipo de excepción.

3. Requisitos no funcionales para la página web.

- Debe aparecer el logo de la Universidad Pública de Navarra en la cabecera de la web.
- Las herramientas del sistema web deben estar en un menú horizontal inmediatamente después de la cabecera.

- El sitio web es privado. Únicamente tienen acceso los responsables de las fincas y los invernaderos de la universidad. Para ello se controlará el acceso mediante un sistema de login y password.
- Se desea que la web del sistema tenga unos colores claros de manera que no resulte pesado para el usuario.
- Estaría bien que en diseño de la página principal tuviera lugar algún tipo de sistema fotográfico con fotos de las fincas, usuarios, redes de sensores ...
- Lo que es la propia página web sería muy recomendable que el diseño estuviera hecho con el gestor de contenidos Joomla.
- Se debe tener muy en cuenta la sincronización de la propia base de datos que usa Joomla con la base de datos del proyecto (temas de sincronización de logins y passwords ...).
- El tamaño de los ficheros subidos por el usuario en con la herramienta Mis Documentos debe ser inferior o igual a 70000 bytes.

2.3 DIAGRAMA ENTIDAD/RELACIÓN



2.3 MODELO RELACIONAL

Continuando con el análisis, continuación voy mostrar el Modelo Relacional (paso a tablas). Dicho modelo sirve para ver la relación entre las tablas de la base de datos. De esta manera se ve como se unen las entidades diseñadas en diagrama E/R.

También se mostrará de qué tipo es cada atributo.

Las claves primarias de las entidades aparecerán subrayadas.

Las claves foráneas, pertenecen a otra entidad aparecerán junto al nombre de la entidad a la hacen referencia entre paréntesis

Usuario

<u>id</u>	<i>int(3)</i>
nombre	<i>varchar(20)</i>
apellidos	<i>varchar(25)</i>
usuario	<i>varchar(20)</i>
contraseña	<i>varchar(25)</i>
tipo	<i>varchar(1)</i>
foto	<i>varchar(60)</i>
correo	<i>varchar(30)</i>
telefono	<i>varchar(10)</i>

Alarma

<u>id_alarma</u>	<i>int(3)</i>
nombre_alarma	<i>varchar(20)</i>
tipo	<i>varchar(15)</i>
signo	<i>varchar(1)</i>
umbral	<i>float</i>
descripción	<i>varchar(100)</i>
accion_correctora	<i>varchar(100)</i>
id (CF -> Usuario)	<i>int(3)</i>

Fichero

<u>id_fichero</u>	<i>int(5)</i>
nombre_fichero	<i>varchar(35)</i>
tamaño_fichero	<i>varchar(30)</i>
tipo_fichero	<i>varchar(10)</i>
id (CF -> Usuario)	<i>int(3)</i>

Red

<u>id_red</u>	<i>int(3)</i>
nombre_red	<i>varchar(20)</i>
descripción	<i>varchar(100)</i>
estructura	<i>longtext</i>

Ubicacion

<u>id_ubi</u>	<i>int(3)</i>
localidad	<i>varchar(20)</i>
zona	<i>varchar(25)</i>
latitud	<i>varchar(10)</i>
longitud	<i>varchar(10)</i>
altitud	<i>varchar(10)</i>

Nodo

<u>id_nodo</u>	<i>int(3)</i>
ip	<i>varchar(4)</i>
mac	<i>varchar(15)</i>
id (CF -> Usuario)	<i>int(3)</i>
id_red (CF -> Red)	<i>int(3)</i>
id_ubi (CF -> Ubicacion)	<i>int(3)</i>

Componente

<u>id_comp</u>	<i>int(3)</i>
nombre_nodo	<i>varchar(15)</i>
id_nodo	<i>int(3)</i>

Medida

<u>id_med</u>	<i>int(5)</i>
tipo	<i>varchar(15)</i>
valor	<i>varchar(20)</i>
fecha	<i>date</i>
hora	<i>varchar(8)</i>
id_comp	<i>int(3)</i>

2.5 DIAGRAMA DE FLUJO DE DATOS

Con este diagrama se pretende visualizar el comportamiento del sistema con todos los elementos externos que afectan al comportamiento del sistema.

**Diagrama de Flujo de Datos
Nivel 0**

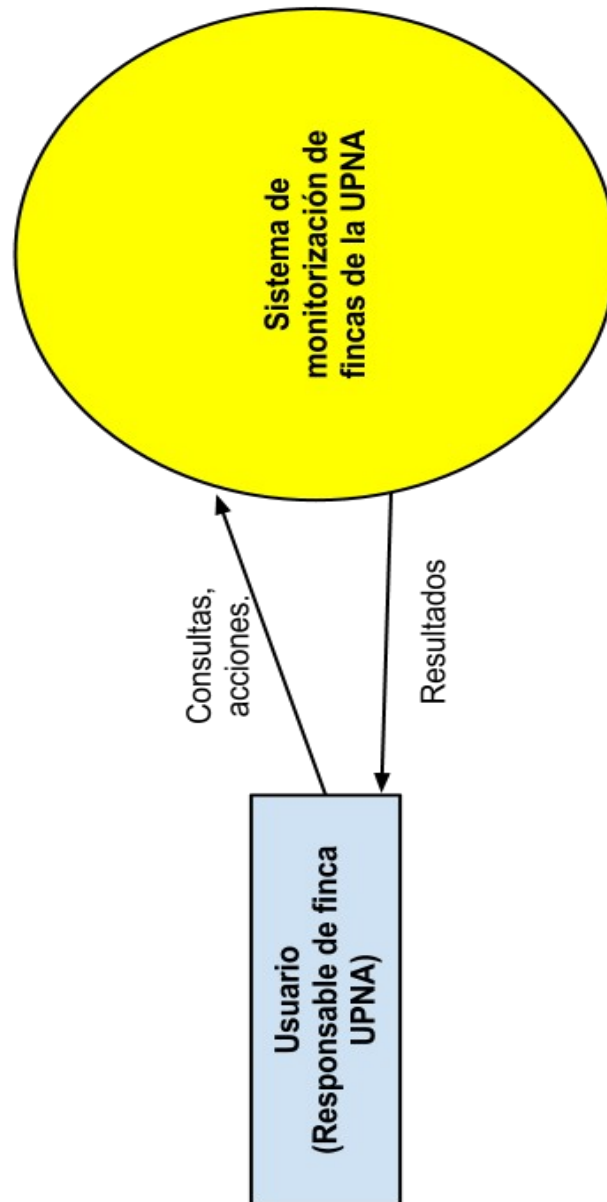
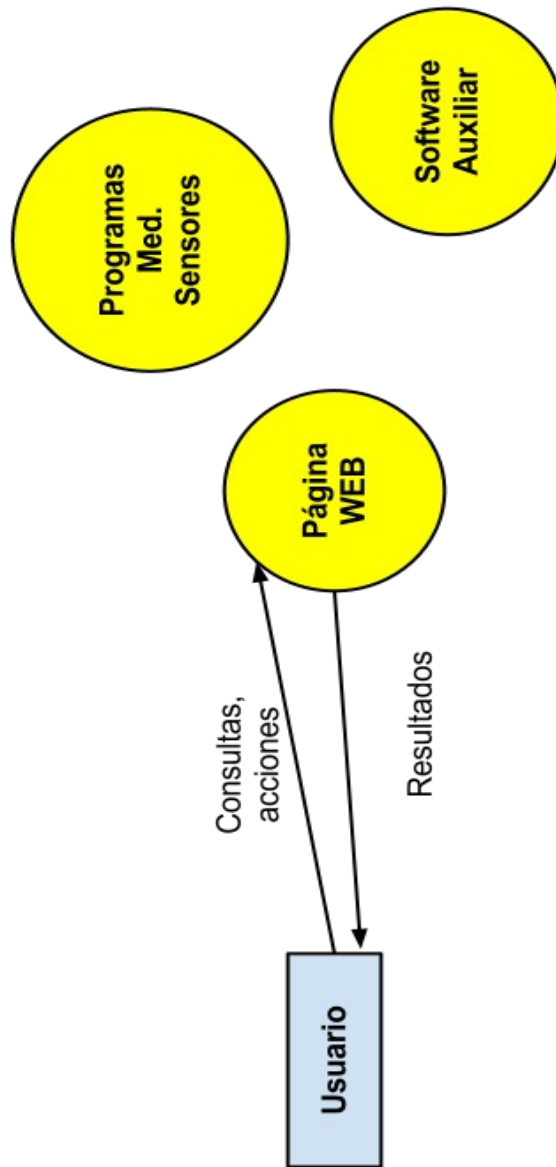
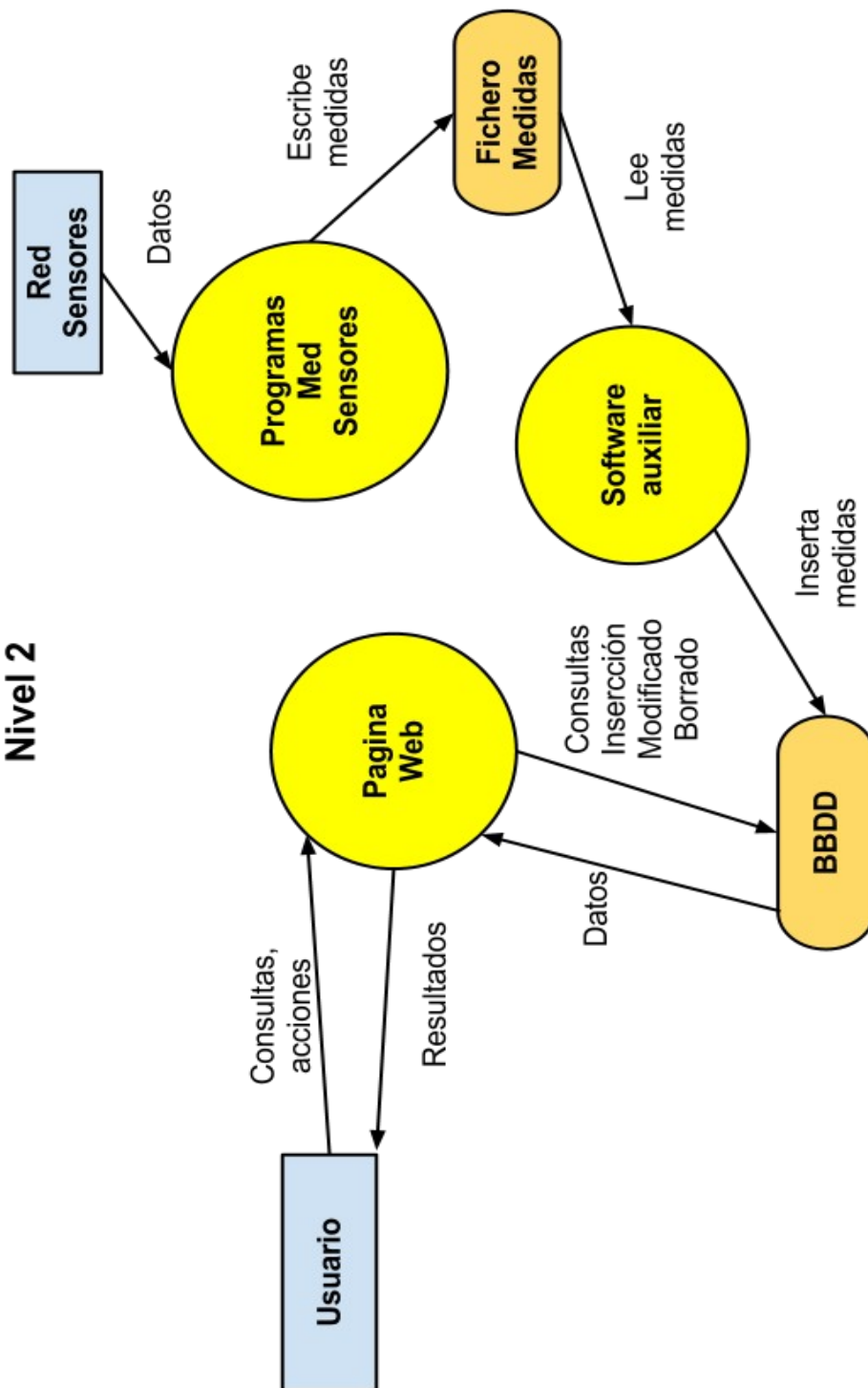


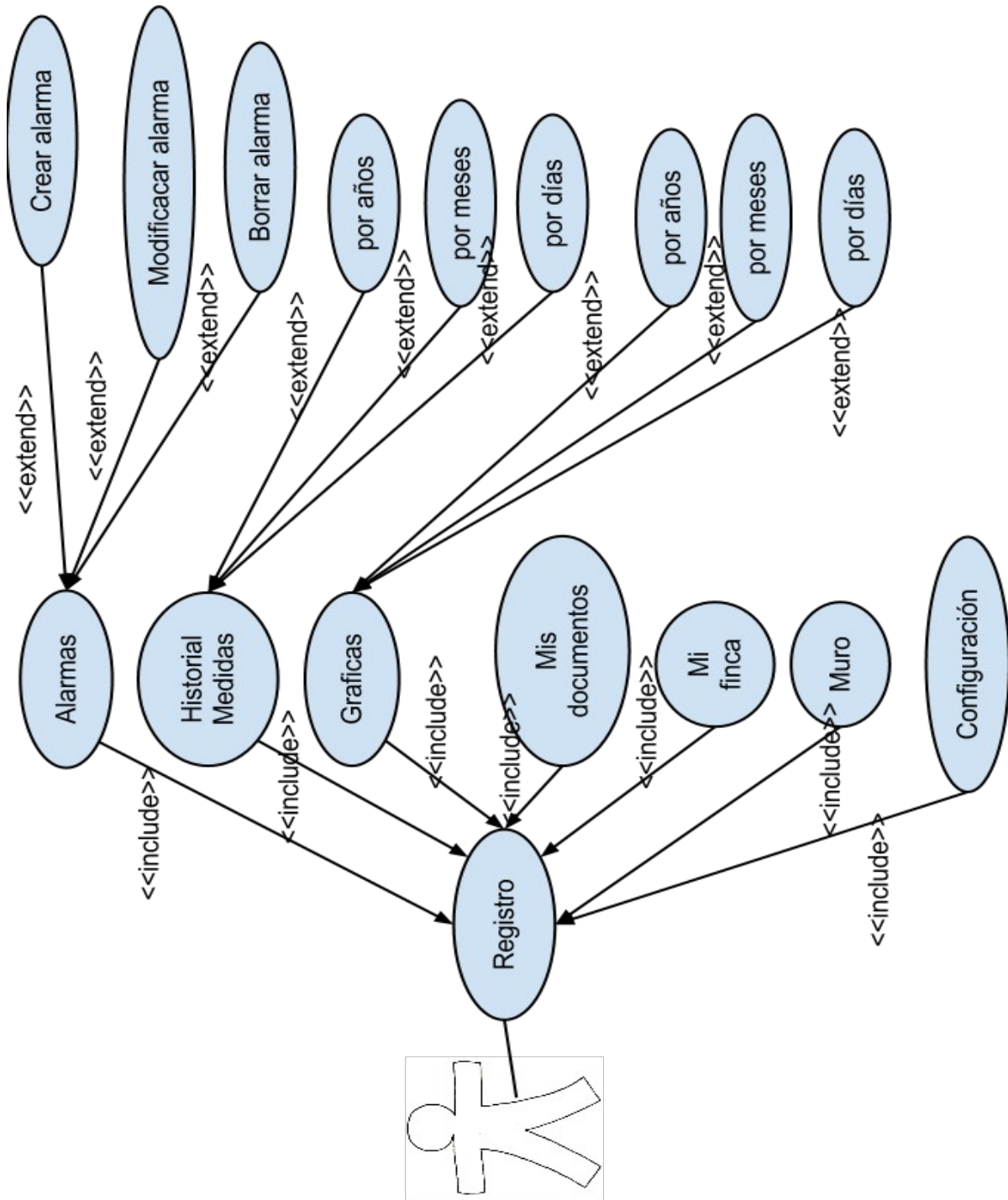
Diagrama de Flujo de Datos Nivel 1



**Diagrama de Flujo de Datos
Nivel 2**

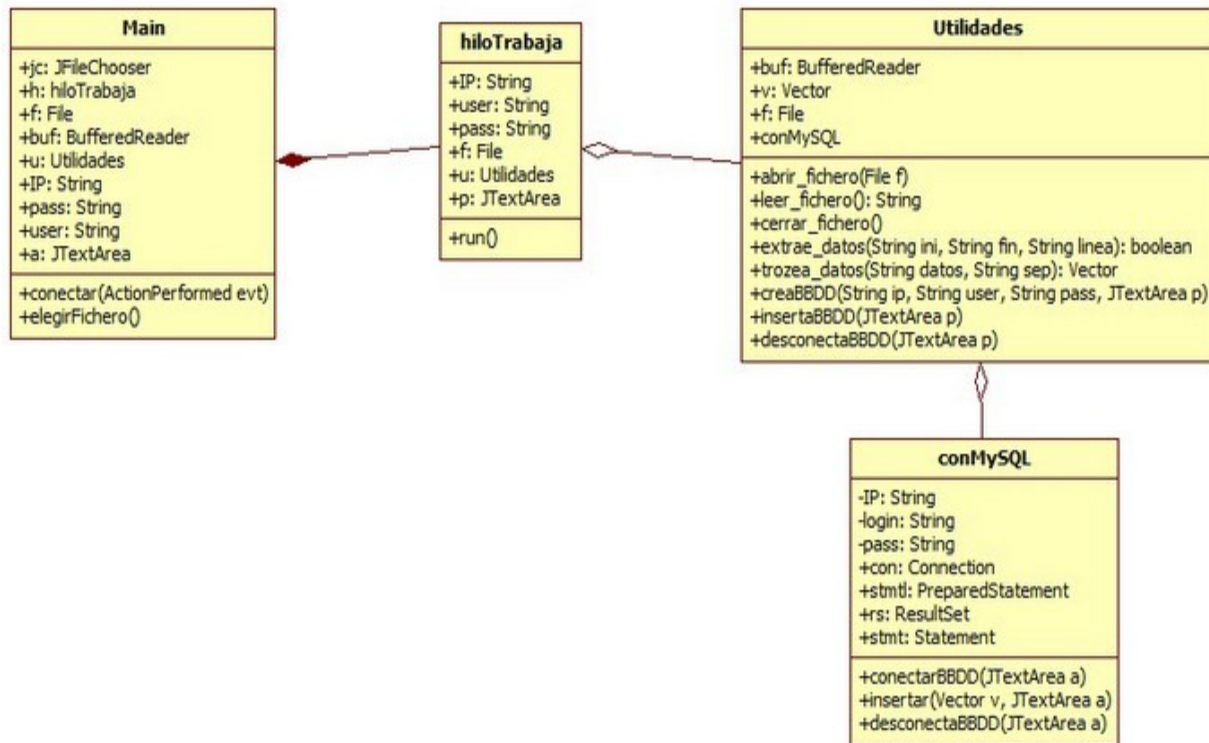


2.6 DIAGRAMA DE CASOS DE USO



2.6 DIAGRAMA DE CLASES

Este va a ser el diagrama de clases correspondiente al software auxiliar, que inserta las medidas tomadas por los dispositivos en la base de datos.



Como se puede observar el hiloTrabaja es la clase que en realidad se va a encargar de realizar el objetivo. Para ello va a usar los métodos de la clase Utilidades, la cual es una agregación de hiloTrabaja.

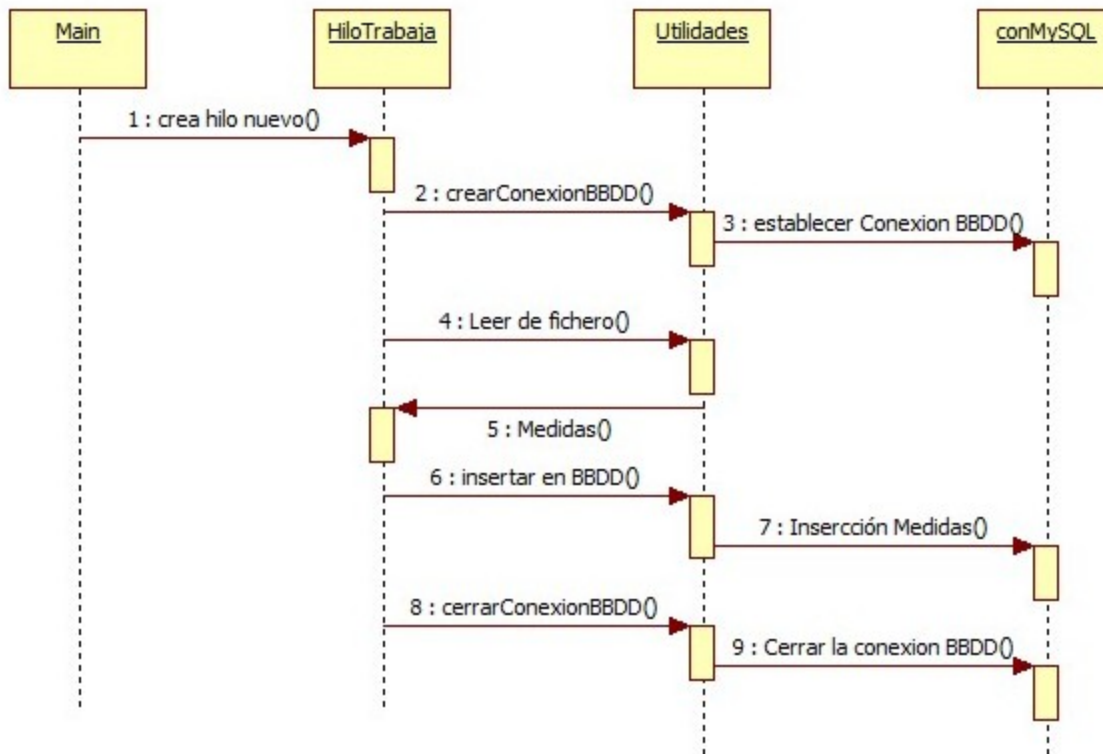
Para que los métodos de la clase Utilidades puedan trabajar correctamente necesita la clase que interactúa con la base de datos, conMySQL, la cual es una agregación de la clase Utilidades.

Por último cabe destacar que la clase hiloTrabaja es una composición de la clase principal, es decir, que en el momento que los objetos de la clase principal sean destruidos también lo harán los objetos de hiloTrabaja. La clase principal, únicamente pide los datos necesarios al usuario y se los pasa al hilo.

2.8 DIAGRAMA DE COLABORACION

En el diagrama de secuencia se puede ver como los objetos interactúan entre sí con el paso del tiempo.

Para este software auxiliar el resultado podría ser el siguiente:



Se ve perfectamente como, con el paso del tiempo (línea discontinua vertical en dirección hacia abajo) el comportamiento del sistema, en las diferentes clase.

Como ya se ha explicado en otros apartados de esta memoria el programa monitoriza el fichero cada cierto tiempo.

En este diagrama de secuencia está representado para una sola iteración. Para que quede más claro.

2.9 PSEUDOCÓDIGO DE LOS MÓDULOS DE LA PÁGINA WEB

No se puede realizar el diagrama de clases ni el diagrama de secuencia para la página web del proyecto ya que, para la programación dinámica de la web se ha usado PHP, un lenguaje no orientado a objetos en un principio.

Así que en lugar de un diagrama de clases mostraré un pseudocódigo de cada herramienta de la web para que se vea su comportamiento.

Primero voy a mostrar un vistazo de lo que sería la web con sus herramientas.



Voy a empezar con la sección del Muro.

Pseudocódigo correspondiente al Muro:

```
<?php
session_start();
//recupero sesion del usuario
$user=$_SESSION["usuario"];
//Me conecto a la base de datos
conexionBBDD();
//consulta e impresion de los datos del usuario
consultar_datos_usuario($user);
echo datos_usuario;
//consulta e impresion de las alarmas del usuario
consultar_alarmas_usuario($user);
echo alarmas_usuario;
?>
```

A continuación voy a mostrar el pseudocódigo de Mis documentos, lugar donde el usuario puede alojar sus documentos.

```
<?php
session_start();
$user=$_SESSION["usuario"];
conexionBBDD();
//recojo los datos de subida del fichero si los hubiese
$nombre_fich=$_POST["nombre"];
...
...
$tamaño=$_POST["tamaño"];
if (usuario_ha_subido_fichero()){
    actualizarFicherosUsuario();
    mover_fichero_a_servidor_web();
    //consulta e impresion de los ficheros del usuario
    consulta_ficheros($user);
    echo ficheros_usuario;
}
else if (borrar_fichero()){
    borrado_del_fichero();//de la bbdd y borrado fisico
    //consulta e impresion de los ficheros del usuario
    consulta_ficheros($user);
    echo ficheros_usuario;
}
else if (descargar_fichero()){
    descarga_fichero_seleccionado();
}
else{//no se ha producido ninguna accion
    //consulta e impresion de los ficheros del usuario
    consulta_ficheros($user);
    echo ficheros_usuario;
}
?>
```

El siguiente apartado es el de alarmas.
Aquí va un posible pseudocódigo:

```
<?php
session_start();
$user=$_SESSION["usuario"];
conexionBBDD();
if (crear_alarma()){
    //recogida datos alarma
    $nombre_alarma=$_POST["nombre"];
    ...
    ...
    $umbral=$_POST["umbral"];
    insertar_nueva_alarma();
    redireccion_muro();
}
else if (modificar_alarma()){
    mostrar_listado_alarmas($user);
    modificar_parametros_alarma_elegida();
}
else if (borrar_alarma()){
    mostrar_listado_alarma($user);
    borrar_alarma_elegida();
}
?>
```

Este sería el pseudocódigo del Historial

```
<?php
session_start();
$user=$_SESSION["usuario"];
conexionBBDD();
//recogida de los intervalos de consulta
$inter_ini=$_POST["ini"];
$inter_fin=$_POST["fin"];
$tabla=consulta_medidas
($inter_ini,$inter_fin,$user);
sincronizacion_med_alarmas($tabla,$user);
/*Impresion de las medidas junto con el nombre
de la alarma que ha hecho dispararse, si fuera
el caso*/
echo medidas;
?>
```

El siguiente apartado es el de gráficas.

```
<?php
session_start();
include 'libreria/grafica';
$user=$_SESSION["usuario"];
conexionBBDD();
//recogida de los intervalos el tipo de grafica
$inter_ini=$_POST["inter_ini"];
$inter_fin=$_POST["inter_fin"];
$tipo_grafico=$_POST["tipo"];
//construccion de la grafica
$estruct=crea_estructura_grafica
($inter_ini,$inter_fin,$user);
//impresion de la estructura
echo imprimir_estructura($estruct);
?>
```

Apartado de visualización de la finca con los dispositivos.

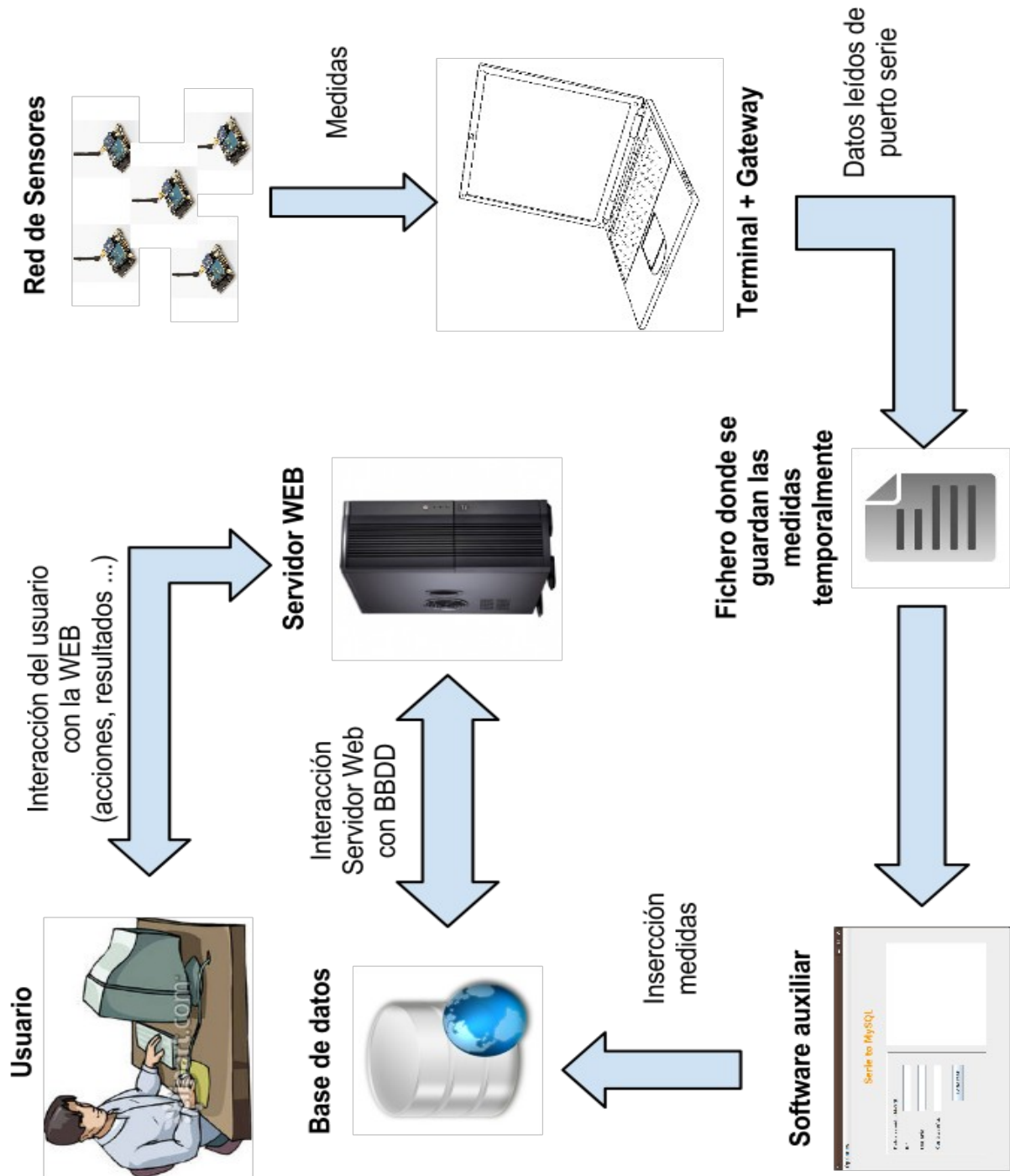
```
<?php
session_start();
$user=$_POST["usuario"];
conexionBBDD();
if (cambios_dibujo_finca()){
    $cambio=$_GET["cambio"];
    actualizo_finca($cambio,$user);
    $finca=consulta_finca($user);
    echo $finca;
}
else {
    /*consulta a la base de datos de la
    extraigo el dibujo en html de la finca
    representada del usuario hasta ese
    momento*/
    $finca=consulta_finca($user);
    echo $finca;
}
?>
```

Por último mostraré el pseudocódigo del panel de configuración del usuario, donde éste podrá cambiar cuando desee sus datos personales (login, pass, foto perfil, teléfono ...).

```
<?php
session_start();
$user=$_SESSION["usuario"];
//me conecto a la base de datos
conexionBBDD();
$form=consulta_datos_personales($user);
//muestro los datos personales en unos
<input type="text">
echo $form;
//recojo los parametros
$login=$_POST["login"];
...
...
$telefono=$_POST["telefono"];
if (hay_cambios()){
    actualiza_cambios();
    redirecciono_a_muro();
}
?>
```

2.10 ARQUITECTURA DEL SISTEMA

Con este apartado se pretende visualizar el comportamiento de todos los elementos que integran el sistema. Ver como interactúan entre ellos y con el usuario final.



2.11 METODOLOGÍA DE TRABAJO

A continuación se va a mostrar la metodología de trabajo que se ha empleado para la realización del proyecto.

Lo primero de todo se planteó una propuesta justificada de proyecto para dar solución al problema planteado. Inmediatamente después se eligieron las tecnologías a usar que se consideraron para alcanzar los objetivos del proyecto.

Después, se pasó a la parte de análisis y diseño, que es la parte clave para que todo el sistema funcione correctamente.

En la parte del análisis se produce la especificación más completa del sistema que se va a implementar (semántica). Justo después la especificación de los requisitos (funcionales y no funcionales).

Luego en la parte del diseño, que es la parte más larga se hacen todos los diagramas pertinentes, diagrama de flujo de datos, de clases, de interacción, el E/R ..., para que el proyectista vaya viendo y modelando perfectamente todo lo necesario para la correcta codificación del siguiente paso.

En esta parte del diseño también se incluye la estimación temporal, en la cual se estima la duración que se tardará en realizar el proyecto, y la arquitectura del sistema, donde se visualiza perfectamente el funcionamiento del sistema.

En el siguiente paso se va a llevar a cabo la implementación. Justo antes se debe elegir el lenguaje o los lenguajes en los que se va a llevar a cabo la implementación del proyecto.

En este apartado en teoría, no se debe pensar cómo resolver conceptos del proyecto, sino que se debe codificar previamente lo que has diseñado en el apartado anterior.

A continuación se van a llevar a cabo las pruebas para comprobar que el sistema funciona correctamente.

Para ello se deben realizar las pruebas de caja negra y caja blanca (unitarias, de carga ...). Las pruebas de caja negra se comprueban metiendo unos datos en el sistema a nivel de usuario y comprobando que los resultados finales que devuelve el sistema son los correctos para cualquier dato de entrada introducido. Se deben probar con todos los casos, sobre todo con los más críticos. En cambio las pruebas de caja blanca evalúan cada parte relevante del sistema por separado introduciendo datos de entrada (incluyendo los más críticos) y comprobando que los resultados son los esperados.

Por último se debe poner toda la parte teórica correctamente detallada, ordenada y explícita en la memoria del proyecto. La parte teórica está formada por todo lo comentado anteriormente, propuesta, análisis, diseño ...

2.12 ESTIMACIÓN TEMPORAL

El siguiente paso a tratar es la estimación temporal del proyecto.

Este es un apartado complejo, en el que hay evaluar cada subapartado de cada parte del desarrollo software del proyecto.

Las cinco partes en las que he dividido el proyecto son:

1. Planteamiento proyecto

Es es la parte en el que la solución del problema a resolver. Tomo las decisiones pertinentes en lo que se refiere a qué lenguaje de programación voy a usar, qué tecnología es más conveniente ... Para ello se debe consultar en diferentes fuentes, que es lo más conveniente.

Para realizar esta tarea estimo que tardaré de 2 a 3 semanas.

2. Análisis y diseño del proyecto

Este apartado, en mi opinión es el apartado más importante del proyecto ya que es donde se van a tomar todas las decisiones del modelado del proyecto. Este apartado se subdivide en dos, análisis y diseño del proyecto.

En el apartado de análisis se detalla muy claramente la especificación del proyecto (semántica), especificación de requisitos (funcionales, no funcionales). Este subapartado de análisis estará en torno a las 3-4 semanas.

El segundo subapartado es el de análisis. Aquí se realizarán todos los diagramas necesarios para visualizar las partes de las que consta el proyecto, tales como diagramas de flujo de datos, diagrama E/R, de casos de uso, de clases ... A la vez en este apartado de diseño se realizará también la metodología de trabajo y la arquitectura del sistema. Todo este apartado del diseño podrá realizarse aproximadamente en un mes o mes y medio.

3. Desarrollo e implementación del proyecto

En este apartado se va a llevar a cabo la codificación, en uno o varios lenguajes de programación, los módulos, clases ... diseñados en el apartado

anterior. Este apartado, si el apartado anterior está bien hecho, no debiera llevar excesivo tiempo. Bien es cierto, que si el proyecto es largo, obviamente habrá bastantes código a implementar. Dicho esto, calculo que para este proyecto la codificación entera del proyecto no debería superar las 3-4 semanas. También supongo que lo que más trabajo, en cuanto la codificación, puede dar es la implementación de la página web.

4. Pruebas

Una vez implementado el proyecto es la hora de hacer las pruebas necesarias (caja negra, caja blanca ...) para evaluar el perfecto funcionamiento del proyecto. Este apartado, en mi opinión es el más impredecible de establecer su duración, ya que resulta imposible evaluar el número y la complejidad de fallos que pudiese haber en la implementación del sistema. Pero por decir algo calculo que en 2 semanas me podría asegurar de que todo funciona correctamente. Pero como ya he dicho el tiempo se puede disparar o acortar.

5. Memoria

En este apartado se debe ordenar, repasar, y aclarar ideas de los puntos 1 2,4. Además debe añadirse la valoración del proyecto, manual de usuarios, anexos ...

Debe quedar todo perfectamente explicado.

Para esta tarea se podría hacer en 3 semanas aproximadamente.

La estimación temporal total, sumando cada una de las partes:

Planteamiento	2,5 semanas
Análisis	3,5 semanas
Diseño	6 semanas
Implementación	3,5 semanas
Pruebas	2 semanas (apartado muy variable)
Memoria	3 semanas

La duración total del proyecto estaría en torno a unas veinte semanas y media, es decir, alrededor de los cinco meses.

3.1 CODIFICACIÓN DE LOS MÓDULOS DE LA DISEÑADOS

Código fuente que gestiona el muro:

```
<?php
    session_start();
    $user=$_SESSION["usuario"];
    //me conecto a la base de datos
    $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
    mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
    //extraigo los datos del usuario
    $select="select nombre,apellidos,correo,telefono,id,foto from Usuario where
usuario='$user'";
    $cons=mysql_query($select,$conectar);
    $datos=mysql_fetch_array($cons);
    $nombre=$datos[0];
    $apellidos=$datos[1];
    $email=$datos[2];
    $telefono=$datos[3];
    $id_user=$datos[4];//para saber las alarmas que tengo que seleccionar. (Mirar la
base de datos. no se si esta bien)
    //me conecto a la base de datos de joomla para extraer el código html del
template
    mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la base de
datos pfcjoomla');
    //extraigo el código de la plantilla murousuario y lo sustituyo por los datos
personales
    $select2="select introtext from pfc_content where alias='murousuario'";
    $consulta=mysql_query($select2,$conectar);
    $template=mysql_fetch_array($consulta);
    $plantilla=str_replace("##foto##",$datos[5],$template[0]);
    $plantilla=str_replace("##nombre##",$nombre,$plantilla);
    $plantilla=str_replace("##apellidos##",$apellidos,$plantilla);
    $plantilla=str_replace("##email##",$email,$plantilla);
    $plantilla=str_replace("##telefono##",$telefono,$plantilla);
    echo $plantilla;
    //extraigo el bloque de las alarmas
    $select2="select introtext from pfc_content where alias='muroalarma'";
    $consulta=mysql_query($select2,$conectar);
```

```

$bloque=mysql_fetch_array($consulta);
//extraigo los datos de las alarmas y
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
$select="select nombre_alarma,tipo,signo,umbral,descripcion,accion_correctora
from Alarma where id=$id_user";
$cons=mysql_query($select,$conectar);
while($datos=mysql_fetch_array($cons)){
    $plantilla=str_replace("##nombre_alarma##",$datos[0],$bloque[0]);
    $plantilla=str_replace("##tipo##",$datos[1],$plantilla);
    $plantilla=str_replace("##signo##",$datos[2],$plantilla);
    $plantilla=str_replace("##umbral##",$datos[3],$plantilla);
    $plantilla=str_replace("##descripcion##",$datos[4],$plantilla);
    $plantilla=str_replace("##accion_correctora##",$datos[5],$plantilla);
    echo $plantilla;
}

?>

```

Código fuente que gestiona el apartado de mis documentos.

```

<?php
    session_start();
    $user=$_SESSION["usuario"];
    include "/var/www/pfc/phps/funciones.php";
    $titulo='<p><strong><span style="color: #0000ff; font-size: large;">Mis
documentos</span></strong></p><br><br>';
    $fichero='<tr><table><td><span style="font-size: medium; color:
#000000;">##fichero##</span></td>
<td align="right" width="77"><a href="descargar.php?
user=##user##&idfich=##idfichero##"></a><a href="index.php/misdocumentos?a=2&idfich=##idfichero##"></a></td>
</tr></table>';
    $subir='<hr /><br><form action="misdocumentos" method="POST"
enctype="multipart/form-data"><span style="font-size: small;"><strong>Subir
fichero:</strong></span> <input id="fichero" type="file" name="fichero" /> <input
type="submit" name="boton" value="Aceptar" /></form>';
    //recojo los datos de subida de fichero (si los hubiese)
    $nombre=$_FILES["fichero"]["name"];

```

```

$tipo=$_FILES["fichero"]["type"];
$tamano=$_FILES["fichero"]["size"];
$boton=$_POST["boton"];
$accion=$_GET["a"];
$idel=$_GET["idfich"];
//me conecto a la base de datos
$conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
//selecciono el id del usuario de la sesion
$seluser="select id from Usuario where usuario='$user'";
$iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
$idusuario=$iduser[0];
if ($boton!=""){//han subido fichero
    $path="/var/www/pfc/ficheros/".$user."/".$nombre;
    //subo el fichero
    move_uploaded_file($_FILES["fichero"]["tmp_name"],$path);
    //lo meto en la base de datos
    $insert="insert into Fichero
values(NULL,'$nombre','$tipo','$tamano','$idusuario)";
    mysql_query($insert);
    //hago la consulta para cargar los ficheros del usuario
    echo $titulo;//imprimo el titulo
    //hago la consulta para cargar los ficheros del usuario
    $sel="select nombre_fichero,id_fichero from Fichero where
id='$idusuario'";
    $cons=mysql_query($sel,$conectar);
    if (mysql_num_rows($cons)>0){
        //echo '<table style="border-width: 0px; border-color: #ffffff; border-
style: solid;" border="0"><tbody>';
        while($datos=mysql_fetch_array($cons)){
            $linea=str_replace("##fichero##",$datos[0],$fichero);
            $linea=str_replace("##idfichero##",$datos[1],$linea);
            $linea=str_replace("##user##",$user,$linea);
            echo $linea;
        }
        //echo '</tbody></table>';
    }
    echo $subir;
}

```

```

}
else if ($accion==2){//borrar
    //doy permisos a todos los ficheros de la carpeta ficheros
    exec("chmod 777 -R /var/www/pfc/ficheros");
    //borro el fichero fisicamente
    $sel="select nombre_fichero from Fichero where id_fichero='$idel'";
    $nombrefichdel=mysql_fetch_array(mysql_query($sel,$conectar));
    $fichdel='rm /var/www/pfc/ficheros/' . $user . '/' . "" . $nombrefichdel[0] . ""';
    exec($fichdel);
    //lo borro de la base de datos
    $delete="delete from Fichero where id_fichero='$idel'";
    mysql_query($delete,$conectar);
    //hago la consulta para cargar los ficheros del usuario
    echo $titulo;//imprimo el titulo
    //hago la consulta para cargar los ficheros del usuario
    $sel="select nombre_fichero,id_fichero from Fichero where
id='$idusuario'";
    $cons=mysql_query($sel,$conectar);
    if (mysql_num_rows($cons)>0){
        //echo '<table style="border-width: 0px; border-color: #ffffff; border-
style: solid;" border="0"><tbody>';
        while($datos=mysql_fetch_array($cons)){
            $linea=str_replace("###fichero###",$datos[0],$fichero);
            $linea=str_replace("###idfichero###",$datos[1],$linea);
            $linea=str_replace("###user###",$user,$linea);
            echo $linea;
        }
        //echo '</tbody></table>';
    }
    echo $subir;
}
else {//situacion del principio solo cargar los ficheros existentes
    echo $titulo;//imprimo el titulo
    //hago la consulta para cargar los ficheros del usuario
    $sel="select nombre_fichero,id_fichero from Fichero where
id='$idusuario'";
    $cons=mysql_query($sel,$conectar);
    if (mysql_num_rows($cons)>0){

```

```

        //echo '<table style="border-width: 0px; border-color: #ffffff; border-
style: solid;" border="0"><tbody>';
        while($datos=mysql_fetch_array($cons)){
            $linea=str_replace("##fichero##",$datos[0],$fichero);
            $linea=str_replace("##idfichero##",$datos[1],$linea);
            $linea=str_replace("##user##",$user,$linea);
            echo $linea;
        }
        //echo '</tbody></table>';
    }
    echo $subir;
}

?>

```

Creación de alarmas:

```

<?php
    session_start();
    $user=$_SESSION["usuario"];
    function valoresCorrectos($nombre,$tipo,$signo,$umbral,$descripcion,
$acc_correc){
        $b=0;
        if ($nombre==""){
            echo "<script language='JavaScript'>alert('ERROR. Es obligatorio
poner un nombre de alarma');</script>";
            $b=1;
        }
        else if (strpos($tipo," "){
            echo "<script language='JavaScript'>alert('ERROR. Es obligatorio
el tipo de alarma');</script>";
            $b=1;
        }
        else if (strpos($signo," "){
            echo "<script language='JavaScript'>alert('ERROR. Es obligatorio
el signo de alarma');</script>";
            $b=1;
        }
        else if ($umbral==""){

```

```

        echo "<script language='JavaScript'>alert('ERROR. Es obligatorio
el valor de alarma');</script>";
        $b=1;
    }
    else if ($descripcion==""){
        echo "<script language='JavaScript'>alert('ERROR. Es obligatorio
una breve descripcion de la alarma');</script>";
        $b=1;
    }
    else if ($acc_correc==""){
        echo "<script language='JavaScript'>alert('ERROR. Es obligatorio
una acción correctora');</script>";
        $b=1;
    }
    return $b;

}
//recojo los valores
$nombre_alarma=mysql_real_escape_string($_POST["nombre_alarma"]);
$tipo=mysql_real_escape_string($_POST["tipo"]);
$signo=mysql_real_escape_string($_POST["signo"]);
$umbral=mysql_real_escape_string($_POST["umbral"]);
$descripcion=mysql_real_escape_string($_POST["descripcion"]);
$acc_correc=mysql_real_escape_string($_POST["acc_correc"]);

//me conecto a la base de datos
$conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
//valores viables introducidos por el usuario
if (valoresCorrectos($nombre_alarma,$tipo,$signo,$umbral,$descripcion,
$acc_correc)==0){
    //compruebo que el nombre de la alarma no esta repetido para un usuario
    $select="select id from Usuario where usuario='$user'";
    $cons=mysql_query($select,$conectar);
    $id=mysql_fetch_array($cons);
    $select="select * from Alarma where nombre_alarma='$nombre_alarma'
and id=$id[0]";

```

```

$cons=mysql_query($select,$conectar);
if (mysql_num_rows($cons)>0){
    echo "<script language='JavaScript'>alert('ERROR. Ya existe una
alarma con ese nombre');</script>";
    mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la
base de datos pfcjoola');
    $select="select introtext from pfc_content where
alias='creaalarma'";
    $consulta=mysql_query($select,$conectar);
    $d=mysql_fetch_array($consulta);
    echo $d[0];
}
else{ //inserto la alarma
    $insert="INSERT INTO Alarma
VALUES(NULL,'$nombre_alarma','$tipo','$signo','$umbral','$descripcion' ,'$acc_correc',
$id[0])";
    mysql_query($insert);
    echo "<script language='JavaScript'>alert('ALARMA creada
correctamente');</script>";
    //muestro el muro del usuario le saldra con la nueva alarma que ha
creado

    //me conecto a la base de datos
    $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die
('Error conectando a mysql');
    mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');

    //extraigo los datos del usuario
    $select="select nombre,apellidos,correo,telefono,id,foto from
Usuario where usuario='$user'";
    $cons=mysql_query($select,$conectar);
    $datos=mysql_fetch_array($cons);
    $nombre=$datos[0];
    $apellidos=$datos[1];
    $email=$datos[2];
    $telefono=$datos[3];
    $id_user=$datos[4];//para saber las alarmas que tengo que
seleccionar. (Mirar la base de datos. no se si esta bien)
    //me conecto a la base de datos de joomla para extraer el código
html del template

```



```

mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la
base de datos pfcjoomla');
//extraigo el código de la plantilla murousuario y lo sustituyo por los
datos personales
$select2="select introtext from pfc_content where
alias='murousuario'";
$consulta=mysql_query($select2,$conectar);
$template=mysql_fetch_array($consulta);
$plantilla=str_replace("##user##",$user,$template[0]);
$plantilla=str_replace("##nombre##",$nombre,$plantilla);
$plantilla=str_replace("##apellidos##",$apellidos,$plantilla);
$plantilla=str_replace("##email##",$email,$plantilla);
$plantilla=str_replace("##telefono##",$telefono,$plantilla);
$plantilla=str_replace("##foto##",$datos[5],$plantilla);
echo $plantilla;
//extraigo el bloque de las alarmas
$select2="select introtext from pfc_content where
alias='muroalarma'";
$consulta=mysql_query($select2,$conectar);
$bloque=mysql_fetch_array($consulta);
//extraigo los datos de las alarmas y
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');
$select="select
nombre_alarma,tipo,signo,umbral,descripcion,accion_correctora from Alarma where
id=$id_user";
$cons=mysql_query($select,$conectar);
while($datos=mysql_fetch_array($cons)){
$plantilla=str_replace("##nombre_alarma##",$datos[0],
$bloque[0]);
$plantilla=str_replace("##tipo##",$datos[1],$plantilla);
$plantilla=str_replace("##signo##",$datos[2],$plantilla);
$plantilla=str_replace("##umbral##",$datos[3],$plantilla);
$plantilla=str_replace("##descripcion##",$datos[4],
$plantilla);
$plantilla=str_replace("##accion_correctora##",$datos[5],
$plantilla);
echo $plantilla;

```

```

    }

    }

}
else{
    mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la base
de datos pfcjoola');
    $select="select introtext from pfc_content where alias='creaalarma'";
    $consulta=mysql_query($select,$conectar);
    $d=mysql_fetch_array($consulta);
    echo $d[0];
}
?>

```

Para el modificado de alarmas:

```

<?php
    session_start();
    $user=$_SESSION["usuario"];
    //
    $boton=$_POST["boton"];
    if (isset($boton)){
        //recojo los parametros
        $id=$_POST["idAlarma"];
        $nombre=$_POST["nombre_alarma"];
        $tipo=$_POST["tipo"];
        $signo=$_POST["signo"];
        $umbral=$_POST["umbral"];
        $descripcion=$_POST["descripcion"];
        $accion_correctora=$_POST["acc_correc"];
        //me conecto a la base de datos de pfc
        $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
        mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de
datos pfc');
        //actualizo la alarma con los valores nuevos
    }
}
?>

```

```

$update="update Alarma set
nombre_alarma='$nombre',tipo='$tipo',signo='$signo',umbral='$umbral',descripcion='$d
descripcion',accion_correctora='$accion_correctora' where id_alarma='$id';
mysql_query($update);
//muestro el listado de las alarmas
$seluser="select id from Usuario where usuario='$user';
$iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
$idusuario=$iduser[0];
//extraigo de la base de datos las el id y nombre de alarma
$select="select id_alarma, nombre_alarma, descripcion from Alarma
where id='$idusuario';
$consulta=mysql_query($select,$conectar);
echo $titulo;
echo '<table style="background-color: #ffffff; border-width: 0px; border-
color: #ffffff; border-style: solid;" border="0"><tbody>';
$prueba='<tr><td style="text-align:
left;"><span><strong><span><span><span style="color: #0000ff; font-size: medium;">-
<span style="color:
#ffffff;">a</span>##nombre_alarma##</span></span></span></strong></span><span
><strong><span><span><span style="color: #ffffff;">a</span><span style="color:
#000000;">##descripcion##</span></span></span></strong></span></td><td
style="text-align: justify;"><a href="index.php/modalarma?
id_alarma=##id_alarma##"></a></td></tr>';
while ($alarma=mysql_fetch_array($consulta)){
    $linea=str_replace("##nombre_alarma##",$alarma[1],$prueba);
    $linea=str_replace("##id_alarma##",$alarma[0],$linea);
    $linea=str_replace("##descripcion##",$alarma[2],$linea);
    echo $linea;
}
echo '</tbody></table>';

}
else {
    $id=$_GET["id_alarma"];
    //me conecto a la base de datos de pfc
    $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');

```

```

mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de
datos pfc');
//extraigo los datos de la alarma a modificar
$select="select
nombre_alarma,tipo,signo,umbral,descripcion,accion_correctora from Alarma where
id_alarma='$id'";
$consulta=mysql_query($select,$conectar);
$alarma=mysql_fetch_array($consulta);
//me conecto a la base de datos pfcjoomla
mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la base
de datos pfcjoomla');
//extraigo el código del template de la plantilla
$select="select introtext from pfc_content where alias='modificaalarma'";
$consulta=mysql_query($select,$conectar);
$d=mysql_fetch_array($consulta);
$linea=str_replace("##id_alarma##",$id,$d[0]);
$linea=str_replace("##nombre_alarma##",$alarma[0],$linea);
$linea=str_replace("##tipo##",$alarma[1],$linea);
$linea=str_replace("##signo##",$alarma[2],$linea);
$linea=str_replace("##umbral##",$alarma[3],$linea);
$linea=str_replace("##descripcion##",$alarma[4],$linea);
$linea=str_replace("##acc_correctora##",$alarma[5],$linea);
echo $linea;
}

?>

```

Para el borrado de alarmas:

```

<?php
session_start();
$user=$_SESSION["usuario"];
$id=$_GET["id_alarma"];
$conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
//borro la alarma
$delete="delete from Alarma where id_alarma='$id'";
mysql_query($delete,$conectar);

```

```

//listo las alarmas restantes
$titulo='<p style="text-align: center;"><span style="font-size: x-large; color:
##ff0000;"><strong>BORRAR ALARMA</strong></span></p><br/>';
$seluser="select id from Usuario where usuario='$user'";
$iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
$idusuario=$iduser[0];
//extraigo de la base de datos las el id y nombre de alarma
$select="select id_alarma, nombre_alarma, descripcion from Alarma where
id='$idusuario'";
$consulta=mysql_query($select,$conectar);
echo $titulo;
echo '<table style="background-color: #ffffff; border-width: 0px; border-color:
#ffffff; border-style: solid;" border="0"><tbody>';
    $prueba='<tr><td style="text-align: left;"><span><strong><span><span><span
style="color: #0000ff; font-size: medium;">-<span style="color:
#ffffff;">a</span>##nombre_alarma##:</span></span></span></strong></span><span
><strong><span><span><span style="color: #ffffff;">a</span><span style="color:
#000000;">##descripcion##</span></span></span></strong></span></td><td
style="text-align: justify;"><a href="index.php/deletealarma?
id_alarma=##id_alarma##"></a></td></tr>';
    while ($alarma=mysql_fetch_array($consulta)){
        $linea=str_replace("##nombre_alarma##",$alarma[1],$prueba);
        $linea=str_replace("##id_alarma##",$alarma[0],$linea);
        $linea=str_replace("##descripcion##",$alarma[2],$linea);
        echo $linea;
    }
    echo '</tbody></table>';
?>

```

Código fuente para el historial por días.

```

<?php
    session_start();
    $user=$_SESSION["usuario"];
    include "/var/www/pfc/libchart/libchart/classes/libchart.php";//librería para los
    graficos
    $opcion1=$_POST["botonFecha"];
    $opcion2=$_POST["botonSensor"];

```

```

//$opcion3=$_GET["listar"];
//$alarm=$_GET["alarm"];
$titulo='<p style="text-align: center;"><span style="font-size: x-large; color:
#ff0000;"><strong>HISTORIAL DE CONSULTAS</strong></span></p><br/>';
$verde="#adff2f";
$rojo="#ff0000";
echo $titulo;
if (isset($opcion1)){//si ha elegido consultar por fecha
    //recojo parametros de fecha
    $diaIni=$_POST["diaini"];
    $mesIni=$_POST["mesini"];
    $anoIni=$_POST["añoini"];
    $diaFin=$_POST["diafin"];
    $mesFin=$_POST["mesfin"];
    $anoFin=$_POST["añoFin"];
    $informe=$_POST["informe"];
    if ((! strstr($diaIni," ")) && (! strstr($mesIni," ")) && (! strstr($anoIni," "))
&& (! strstr($diaFin," ")) && (! strstr($mesFin," ")) && (! strstr($anoFin," "))) {
        $fechalni=$anoIni."/".$mesIni."/".$diaIni;
        $_SESSION["fechainidia"]=$fechalni;
        $fechaFin=$anoFin."/".$mesFin."/".$diaFin;
        $_SESSION["fechafindia"]=$fechaFin;
        //me conecto a la base de datos
        $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die
('Error conectando a mysql');
        mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');
        $seluser="select id from Usuario where usuario='$user'";
        $iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
        $idusuario=$iduser[0];
        $select="select m.id_med, m.tipo, m.valor, m.fecha, m.hora,
m.id_comp from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fechalni' and
'$fechaFin' order by m.tipo,m.fecha,m.hora";
        $consulta=mysql_query($select,$conectar);
        //cojo las alarmas
        $selalarma="select signo, umbral,tipo,nombre_alarma from Alarma
where id='$idusuario'";

```

```

mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la
base de datos pfcjoomla');
$select2="select introtext from pfc_content where
alias='plantillahistorial'";
$consulta2=mysql_query($select2,$conectar);
$dato=mysql_fetch_array($consulta2);
$template=$dato[0];
$cont=0;
$ex=0;
if (isset($informe)){
    $fd=fopen("/var/www/pfc/ficheros/informes/informe.doc","w");
    if (stristr($fechalni,$fechaFin)){
        fputs($fd,"Informe del día ".$fechalni."\n\n");
    }
    else{
        fputs($fd,"Informe entre los días ".$fechalni." y ".
$fechaFin."\n\n");
    }

    fputs($fd,"\nNº\tCaracterística\t\tValor\tFecha\t\tHora\t\tNodo\t\tAlarma\n\n");
    $ex=1;
    $al="-";
}
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');
if (mysql_num_rows($consulta)>0){
    while($tabla=mysql_fetch_array($consulta)){
        $cont++;
        $linea=str_replace("##id_med##",$cont,$template);
        $linea=str_replace("##tipo##",$tabla[1],$linea);
        $linea=str_replace("##valor##",$tabla[2],$linea);
        $linea=str_replace("##fecha##",$tabla[3],$linea);
        $linea=str_replace("##hora##",$tabla[4],$linea);
        $linea=str_replace("##item##","historialalarma",
$linea);

        $selaux="select n.ip from Nodo n,Componente c,
Medida m where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
c.id_comp='$tabla[5]' group by n.ip";

```

```

$consaux=mysql_query($selaux,$conectar);
$nodo=mysql_fetch_array($consaux);
$linea=str_replace("##nodo##",$nodo[0],$linea);
//veo a ver de que color pinto la linea
$consalarma=mysql_query($selalarma,$conectar);
while($t=mysql_fetch_array($consalarma)){
    if (strstr($tabla[1],$t[2])){
        if (strstr($t[0],"<")){
            if ($tabla[2]<$t[1]){

$linea=str_replace("##color##",$rojo,$linea);

$linea=str_replace("##n_alarma##",$t[3],$linea);

                                $al=$t[3];
                                }
                            }
                        else if (strstr($t[0],">")){
                            if ($tabla[2]>$t[1]){

$linea=str_replace("##color##",$rojo,$linea);

$linea=str_replace("##n_alarma##",$t[3],$linea);

                                $al=$t[3];
                                }
                            }
                        else if (strstr($t[0], "=")){
                            if ($tabla[2]==$t[1]){

$linea=str_replace("##color##",$rojo,$linea);

$linea=str_replace("##n_alarma##",$t[3],$linea);

                                $al=$t[3];
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
$linea=str_replace("##color##",$verde,$linea);
$linea=str_replace("##n_alarma##","", $linea);
$linea=str_replace("##b##","1", $linea);

```



```

        echo $linea;
        if ($ex==1){
            if ((stristr($tabla[1],"CO2")) ||
(stristr($tabla[1],"Ozono"))){
                fputs($fd,$cont."\t".$tabla[1]."\t\t\t".
$tabla[2]."\t".$tabla[3]."\t".$tabla[4]."\t".$nodo[0]."\t".$al."\n");
            }
            else{
                fputs($fd,$cont."\t".$tabla[1]."\t\t\t".
$tabla[2]."\t".$tabla[3]."\t".$tabla[4]."\t".$nodo[0]."\t".$al."\n");
            }
            $al="-";
        }
    }
    /*ahora voy a imprimir la media, max, min de cada uno de
los sensores en el dia*/
    //1º cojo los sensores de la fecha
    $select="select m.tipo from Medida m,Componente c,Nodo
n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaIni' and '$fechaFin' group by m.tipo";
    $consulta=mysql_query($select,$conectar);
    while($sensores=mysql_fetch_array($consulta)){//para cada
uno de los sensores calculo max,min,avg
        $maxmedia="select
avg(m.valor),max(m.valor),min(m.valor) from Medida m,Componente c,Nodo n where
m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha
between '$fechaIni' and '$fechaFin' and m.tipo='$sensores[0]' order by m.fecha";
        $consmaxmedia=mysql_query($maxmedia,
$conectar);

        $datosmaxmedia=mysql_fetch_array($consmaxmedia);
        if ($datosmaxmedia[2]<0){
            $min="select m.valor from Medida
m,Componente c,Nodo n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
n.id='$idusuario' and m.fecha between '$fechaIni' and '$fechaFin' and
m.tipo='$sensores[0]' and m.valor<0 group by m.valor order by valor desc";
            $consmin=mysql_query($min,$conectar);
            $datosmin=mysql_fetch_array($consmin);

```

```

        $minimo=$datosmin[0];
    }
    else{
        $minimo=$datosmaxmedia[2];
    }
    $plant='<br><table style="background-color: #f0ffff;"
border="0"><tr><td><p><span style="color: #0000ff; font-size:
medium;"><strong>Datos globales: '.$sensores[0].'/</strong></span></p><p></p>
<p><span style="font-size: small;"><strong><span style="color: #000000;">Medida
máxima:</span></strong> '.$datosmaxmedia[1].'          <strong><span
style="color: #000000;">Medida mínima</span>:</strong> '.$minimo.'
<strong><span style="color: #000000;">Media:</span></strong> '.
$datosmaxmedia[0].'/</span></p></td></tr></table>';
    echo $plant;
    if ($ex==1){
        fputs($fd,"\nDatos Globales ".
$sensores[0]."\n"."Medida Maxima: ".$datosmaxmedia[1]." Medida Mínima: ".$minimo."
Media: ".$datosmaxmedia[0]."\n\n");
    }

}
if ($ex==1){
    fclose($fd);
    $ex=0;
    header("Location: http://localhost/pfc/descargar.php?
flagf=1");
}
}
else {
    header("Location: historial");
    echo "<script language='JavaScript'>alert('NO hay
coincidencias en la consulta');</script>";
}
}
else{
    header("Location: historial");
}
}
?>

```

Código fuente para el historial por meses y años.

```
<?php
    session_start();
    $user=$_SESSION["usuario"];
    $opcion3=$_POST["botonAno"];
    $opcion4=$_POST["botonMes"];
    $titulo='<p style="text-align: center;"><span style="font-size: x-large; color:
#ff0000;"><strong>HISTORIAL DE CONSULTAS</strong></span></p><br/>';
    $verde="#adff2f";
    $rojo="#ff0000";
    echo $titulo;
    if (isset($opcion3)){
        $anoIni=$_POST["añoini"];
        $anoFin=$_POST["añoFin"];
        $informe=$_POST["informe"];
        $_SESSION["añoiniaux"]=$anoIni;
        $_SESSION["añoфинаux"]=$anoFin;
        if ((! strstr($anoIni," ") && (! strstr($anoFin," "))) {
            $fechaini=$anoIni."/01/01";
            $fechafin=$anoFin."/12/31";
            $_SESSION["fechainiaño"]=$fechaini;
            $_SESSION["fechafinaño"]=$fechafin;
            //me conecto a la base de datos
            $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die
('Error conectando a mysql');
            mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');

            $seluser="select id from Usuario where usuario='$user'";
            $iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
            $idusuario=$iduser[0];
            $select="select m.id_med, m.tipo, m.valor, m.fecha, m.hora,
m.id_comp from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fechaini' and
'$fechafin' order by m.tipo, m.fecha, m.hora";
            $consulta=mysql_query($select,$conectar);
            //cojo las alarmas
```

```

        $selalarma="select signo, umbral, tipo, nombre_alarma from Alarma
where id='$idusuario'";
        mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la
base de datos pfcjoomla');
        $select2="select introtext from pfc_content where
alias='plantillahistorial'";
        $consulta2=mysql_query($select2,$conectar);
        $dato=mysql_fetch_array($consulta2);
        $template=$dato[0];
        //$linea=str_replace("##item##","historialalarmabis",$template);
        $cont=0;
        $ex=0;
        if (isset($informe)){
            $fd=fopen("/var/www/pfc/ficheros/informes/informe.doc","w");
            if (strpos($fechaini,$fechafin)){
                fputs($fd,"Informe del día ".$fechaini."\n\n");
            }
            else{
                fputs($fd,"Informe entre los días ".$fechaini." y ".
$fechafin."\n\n");
            }

            fputs($fd,"\nNº\tCaracterística\t\tValor\tFecha\t\tHora\tNodo\tAlarma\n\n");
            $ex=1;
            $al="-";
        }
        mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');
        if (mysql_num_rows($consulta)>0){
            while($tabla=mysql_fetch_array($consulta)){
                $cont++;
                $linea=str_replace("##id_med##",$cont,$template);
                $linea=str_replace("##tipo##",$tabla[1],$linea);
                $linea=str_replace("##valor##",$tabla[2],$linea);
                $linea=str_replace("##fecha##",$tabla[3],$linea);
                $linea=str_replace("##hora##",$tabla[4],$linea);
                $linea=str_replace("##item##","historialalarmabis",
$linea);

```

```

        $selaux="select n.ip from Nodo n,Componente c,
Medida m where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
c.id_comp='$tabla[5]' group by n.ip";
        $consaux=mysql_query($selaux,$conectar);
        $nodo=mysql_fetch_array($consaux);
        $linea=str_replace("###nodo##",$nodo[0],$linea);
        //veo a ver de que color pinto la linea
        $consalarma=mysql_query($selalarma,$conectar);
        while($t=mysql_fetch_array($consalarma)){
            if (strstr($tabla[1],$t[2])){
                if (strstr($t[0],"<")){
                    if ($tabla[2]<$t[1]){

$linea=str_replace("###color##",$rojo,$linea);

$linea=str_replace("###n_alarma##",$t[3],$linea);

                                $al=$t[3];
                                }
                            }
                        else if (strstr($t[0],">")){
                            if ($tabla[2]>$t[1]){

$linea=str_replace("###color##",$rojo,$linea);

$linea=str_replace("###n_alarma##",$t[3],$linea);

                                $al=$t[3];
                                }
                            }
                        else if (strstr($t[0], "=")){
                            if ($tabla[2]==$t[1]){

$linea=str_replace("###color##",$rojo,$linea);

$linea=str_replace("###n_alarma##",$t[3],$linea);

                                $al=$t[3];
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

$linea=str_replace("##color##",$verde,$linea);
$linea=str_replace("##n_alarma##","",$linea);
$linea=str_replace("##b##","3",$linea);
echo $linea;
if ($ex==1){
    if ((stristr($tabla[1],"CO2")) ||
(stristr($tabla[1],"Ozono"))){
        fputs($fd,$cont."\t".$tabla[1]."\t\t\t".
$tabla[2]."\t".$tabla[3]."\t".$tabla[4]."\t".$nodo[0]."\t".$al."\n");
    }
    else{
        fputs($fd,$cont."\t".$tabla[1]."\t\t\t".
$tabla[2]."\t".$tabla[3]."\t".$tabla[4]."\t".$nodo[0]."\t".$al."\n");
    }
    $al="-";
}
}
/*ahora voy a imprimir la media, max, min de cada uno de
los sensores en el dia*/
//1º cojo los sensores de la fecha
$select="select m.tipo from Medida m,Componente c,Nodo
n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaini' and '$fechafin' group by m.tipo";
$consulta=mysql_query($select,$conectar);
while($sensores=mysql_fetch_array($consulta)){//para cada
uno de los sensores calculo max,min,avg
    $maxmedia="select
avg(m.valor),max(m.valor),min(m.valor) from Medida m,Componente c,Nodo n where
m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha
between '$fechaini' and '$fechafin' and m.tipo='$sensores[0]' order by m.fecha";
    $consmaxmedia=mysql_query($maxmedia,
$conectar);

    $datosmaxmedia=mysql_fetch_array($consmaxmedia);
    if ($datosmaxmedia[2]<0){
        $min="select m.valor from Medida
m,Componente c,Nodo n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and

```

```

n.id='$idusuario' and m.fecha between '$fechaini' and '$fechafin' and
m.tipo='$sensores[0]' and m.valor<0 group by m.valor order by valor desc";
        $consmin=mysql_query($min,$conectar);
        $datosmin=mysql_fetch_array($consmin);
        $minimo=$datosmin[0];
    }
    else{
        $minimo=$datosmaxmedia[2];
    }
    $plant='<br><table style="background-color: #f0ffff;"
border="0"><tr><td><p><span style="color: #0000ff; font-size:
medium;"><strong>Datos globales: '.$sensores[0].'/></strong></span></p><p></p>
<p><span style="font-size: small;"><strong><span style="color: #000000;">Medida
máxima:</span></strong> '.$datosmaxmedia[1].'                <strong><span
style="color: #000000;">Medida mínima</span></strong> '.$minimo.'
<strong><span style="color: #000000;">Media:</span></strong> '.
$datosmaxmedia[0].'/></span></p></td></tr></table>';
    echo $plant;
    if ($ex==1){
        fputs($fd,"\nDatos Globales ".
$sensores[0]."\n"."Medida Maxima: ".$datosmaxmedia[1]."\n Medida Mínima: ".$minimo."
Media: ".$datosmaxmedia[0]."\n\n");
    }
}
if ($ex==1){
    fclose($fd);
    $ex=0;
    header("Location: http://localhost/pfc/descargar.php?
flagf=1");
}
}
else {
    header("Location: historial");
    echo "<script language='JavaScript'>alert('NO hay
coincidencias en la consulta');</script>";
}
}
else{
    header("Location: historial");
}

```

```

    }
}
else if (isset($opcion4)){
    $mesIni=$_POST["MIni"];
    $anoIni=$_POST["MAIni"];
    $mesFin=$_POST["MFin"];
    $anoFin=$_POST["MAFin"];
    $informe=$_POST["informe"];
    if ((! strstr($mesIni," ")) && (! strstr($anoIni," ")) && (! strstr($mesFin," "))
&& (! strstr($anoFin," ")){
        $fechaini=$anoIni."/".$mesIni."/01";
        $fechafin=$anoFin."/".$mesFin."/31";
        $_SESSION["fechainimes"]=$fechaini;
        $_SESSION["fechafinmes"]=$fechafin;
        //me conecto a la base de datos
        $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die
('Error conectando a mysql');
        mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');
        $seluser="select id from Usuario where usuario='$user'";
        $iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
        $idusuario=$iduser[0];
        $select="select m.id_med, m.tipo, m.valor, m.fecha, m.hora,
m.id_comp from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fechaini' and
'$fechafin' order by m.tipo, m.fecha, m.hora";
        $consulta=mysql_query($select,$conectar);
        //cojo las alarmas
        $selalarma="select signo, umbral,tipo,nombre_alarma from Alarma
where id='$idusuario'";
        mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la
base de datos pfcjoomla');
        $select2="select introtext from pfc_content where
alias='plantillahistorial'";
        $consulta2=mysql_query($select2,$conectar);
        $dato=mysql_fetch_array($consulta2);
        $template=$dato[0];
        //$linea=str_replace("##item##","historialalarmabis",$template);
        $cont=0;

```



```

$ex=0;
if (isset($informe)){
    $fd=fopen("/var/www/pfc/ficheros/informes/informe.doc","w");
    if (strpos($fechaini,$fechafin)){
        fputs($fd,"Informe del día ".$fechaini."\n\n");
    }
    else{
        fputs($fd,"Informe entre los días ".$fechaini." y ".
$fechafin."\n\n");
    }

    fputs($fd,"\nNº\tCaracterística\t\tValor\tFecha\t\tHora\t\tNodo\t\tAlarma\n\n");
    $ex=1;
    $al="-";
}
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base
de datos pfc');
if (mysql_num_rows($consulta)>0){
    while($tabla=mysql_fetch_array($consulta)){
        $cont++;
        $linea=str_replace("##id_med##",$cont,$template);
        $linea=str_replace("##tipo##",$tabla[1],$linea);
        $linea=str_replace("##valor##",$tabla[2],$linea);
        $linea=str_replace("##fecha##",$tabla[3],$linea);
        $linea=str_replace("##hora##",$tabla[4],$linea);
        $linea=str_replace("##item##","historialalarmabis",
$linea);

        $selaux="select n.ip from Nodo n,Componente c,
Medida m where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
c.id_comp='$tabla[5]' group by n.ip";
        $consaux=mysql_query($selaux,$conectar);
        $nodo=mysql_fetch_array($consaux);
        $linea=str_replace("##nodo##",$nodo[0],$linea);
        //veo a ver de que color pinto la linea
        $consalarma=mysql_query($selalarma,$conectar);
        while($t=mysql_fetch_array($consalarma)){
            if (strpos($tabla[1],$t[2])){
                if (strpos($t[0],"<")){

```

```

        if ($tabla[2]<$t[1]){

$linea=str_replace("##color##",$rojo,$linea);

$linea=str_replace("##n_alarma##",$t[3],$linea);

        $al=$t[3];
        }
    }
    else if (stristr($t[0],">")){
        if ($tabla[2]>$t[1]){

$linea=str_replace("##color##",$rojo,$linea);

$linea=str_replace("##n_alarma##",$t[3],$linea);

        $al=$t[3];
        }
    }

    else if (stristr($t[0], "=")){
        if ($tabla[2]==$t[1]){

$linea=str_replace("##color##",$rojo,$linea);

$linea=str_replace("##n_alarma##",$t[3],$linea);

        $al=$t[3];
        }
    }
}

$linea=str_replace("##color##",$verde,$linea);
$linea=str_replace("##n_alarma##","", $linea);
$linea=str_replace("##b##", "4", $linea);
echo $linea;
if ($ex==1){
    if ((stristr($tabla[1], "CO2")) ||
(stristr($tabla[1], "Ozono"))){
        fputs($fd,$cont."\t".$tabla[1]."\t\t\t".
$tabla[2]."\t".$tabla[3]."\t".$tabla[4]."\t".$nodo[0]."\t".$al."\n");
    }
}

```

```

else{
    fputs($fd,$cont."\t".$tabla[1]."\t\t".
$tabla[2]."\t".$tabla[3]."\t".$tabla[4]."\t".$nodo[0]."\t".$al."\n");
}
$al="-";
}
}
/*ahora voy a imprimir la media, max, min de cada uno de
los sensores en el dia*/
//1º cojo los sensores de la fecha
$select="select m.tipo from Medida m,Componente c,Nodo
n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaini' and '$fechafin' group by m.tipo";
$consulta=mysql_query($select,$conectar);
while($sensores=mysql_fetch_array($consulta)){//para cada
uno de los sensores calculo max,min,avg
$maxmedia="select
avg(m.valor),max(m.valor),min(m.valor) from Medida m,Componente c,Nodo n where
m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha
between '$fechaini' and '$fechafin' and m.tipo='$sensores[0]' order by m.fecha";
$consmmaxmedia=mysql_query($maxmedia,
$conectar);

$datosmaxmedia=mysql_fetch_array($consmmaxmedia);
if ($datosmaxmedia[2]<0){
    $min="select m.valor from Medida
m,Componente c,Nodo n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
n.id='$idusuario' and m.fecha between '$fechaini' and '$fechafin' and
m.tipo='$sensores[0]' and m.valor<0 group by m.valor order by valor desc";
    $consmin=mysql_query($min,$conectar);
    $datosmin=mysql_fetch_array($consmin);
    $minimo=$datosmin[0];
}
else{
    $minimo=$datosmaxmedia[2];
}
$plant='<br><table style="background-color: #f0ffff;"
border="0"><tr><td><p><span style="color: #0000ff; font-size:
medium;"><strong>Datos globales: '.$sensores[0].'/</strong></span></p><p></p>

```

```

<p><span style="font-size: small;"><strong><span style="color: #000000;">Medida
máxima:</span></strong> '.$datosmaxmedia[1].'          <strong><span
style="color: #000000;">Medida mínima:</span></strong> '.$minimo.'
<strong><span style="color: #000000;">Media:</span></strong> '.
$datosmaxmedia[0].</span></p></td></tr></table>';
        echo $plant;
        if ($ex==1){
            fputs($fd,"nDatos Globales ".
$sensores[0]."\n"."Medida Maxima: ".$datosmaxmedia[1]." Medida Mínima: ".$minimo."
Media: ".$datosmaxmedia[0]."\n\n");
        }
    }
    if ($ex==1){
        fclose($fd);
        $ex=0;
        header("Location: http://localhost/pfc/descargar.php?
flagf=1");
    }
}
else {
    header("Location: historial");
    echo "<script language=JavaScript>alert('NO hay
coincidencias en la consulta');</script>";
}
}
else{
    header("Location: historial");
}
}
?>

```

A continuación debería poner el código fuente de los script que se encargan de refrescar las medidas junto con las características de la alarma que ha pinchado el usuario. Pero básicamente es el mismo código más una consulta de dicha alarma. Así que lo voy a omitir.

Este es el código que gobierna, junto con la librería Libchart el apartado de Gráficos.

```

<?php
    session_start();
    $user=$_SESSION["usuario"];
    include "/var/www/pfc/libchart/libchart/classes/libchart.php";
    include "/var/www/pfc/phps/funciones.php";
    $opcion3=$_POST["botonanograp"];
    $opcion4=$_POST["botonmesgraf"];
    $opcion1=$_POST["botondiagraf"];
    //me conecto a la base de datos
    $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
    mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
    //selecciono el id del usuario
    $seluser="select id from Usuario where usuario='$user'";
    $iduser=mysql_fetch_array(mysql_query($seluser,$conectar));
    $idusuario=$iduser[0];
    if (isset($opcion3)){
        $anoIni=$_POST["añoini"];
        $anoFin=$_POST["añoFin"];
        $tipoGrafico=$_POST["tipograp"];
        if ((! strstr($anoIni," ")) && (! strstr($anoFin," "))){
            $fechaini=$anoIni."/01/01";
            $fechafin=$anoFin."/12/31";
            $select="select m.tipo from Medida m,Componente c,Nodo n
where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaini' and '$fechafin' group by m.tipo";
            $consulta=mysql_query($select,$conectar);
            while($sensores=mysql_fetch_array($consulta)){
                $bool=0;
                $a=$anoFin-$anoIni;
                while($bool==0){
                    $ano=$anoFin-$a;
                    $chart=tipo_grafico($tipoGrafico);
                    $dataSet = new XYDataSet();
                    $dataSetmax = new XYDataSet();
                    $dataSetmin = new XYDataSet();
                    echo "<strong><font color='red'>Maximo <font
color='blue'>Minimo <font color='orange'>Media <font color='black'>de ".
$sensores[0]." de los meses de ".$ano."</strong>";

```

```

echo "<br><br>";
for ($i=1;$i<=12;$i++){
    $fini=$ano."/".$i."/01";
    $ffin=$ano."/".$i."/31";
    $s="select avg(m.valor), max(m.valor),
min(m.valor) from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fini' and '$ffin' and
m.tipo='$sensores[0]' order by m.fecha";
    $c=mysql_query($s,$conectar);
    while ($g=mysql_fetch_array($c)){
        $m=transforma_mes($i);
        if ($g[0]==""){
            $media=0;
            $mediaux="-";
        }
        else{
            $media=$g[0];
            $mediaux=$g[0];
        }
        if ($g[1]==""){
            $max="-";
        }
        else{
            $max=$g[1];
        }
        if ($g[2]==""){
            $min="-";
        }
        else{
            if ($g[2]<0){
                $minsel="select m.valor
from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fechaini' and
'$fechafin' and m.tipo='$sensores[0]' and m.valor<0 group by m.valor order by valor
desc";
                $consmin=mysql_query($minsel,$conectar);
                $datosmin=mysql_fetch_array($consmin);
            }
        }
    }
}

```

```

$min=$datosmin[0];
    }
    else{
        $min=$g[2];
    }
}
$dataSet->addPoint(new Point($m,
$media));
$dataSetmax->addPoint(new Point($m,
$max));
$dataSetmin->addPoint(new Point($m,
$min));
    }
    echo "<font
color=\"black\">".transforma_mes($i)." <font color=\"red\">".$max." <font
color=\"blue\">".$min." <font color=\"orange\">".$mediaux;
    echo "<br>";
}
$globaldata=new XYSeriesDataSet();
$globaldata->addSerie("Media",$dataSet);
$globaldata->addSerie("Máxima",$dataSetmax);
$globaldata->addSerie("Mínima",$dataSetmin);
$chart->setDataSet($globaldata);
$chart->setTitle("Medidas de ".$sensores[0]." medias
de los meses de ".$ano);
$chart->render("/var/www/pfc/images/graficos/"
$sensores[0].$ano.".png");
echo '<p></p>';
    if ($a==0){
        $bool=1;
    }
    $a--;
}
echo "<p align=\"center\" ><font
color=\"black\">-----
-----</p><br>";
    }
}
}

```

```

else {
    header("Location: graficas");
}
}
else if (isset($opcion4)){
    $mesIni=$_POST["MIni"];
    $anoIni=$_POST["MAIni"];
    $mesFin=$_POST["MFin"];
    $anoFin=$_POST["MAFin"];
    $tipoGrafico=$_POST["tipograf"];
    if ((! strstr($mesIni," ")) && (! strstr($anoIni," ")) && (! strstr($mesFin," "))
&& (! strstr($anoFin," "))) {
        $fechaini=$anoIni."/".$mesIni."/01";
        $fechafin=$anoFin."/".$mesFin."/31";
        $select="select m.tipo from Medida m,Componente c,Nodo n
where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaini' and '$fechafin' group by m.tipo";
        $consulta=mysql_query($select,$conectar);
        while($sensores=mysql_fetch_array($consulta)){
            $a=$anoFin-$anoIni;
            if ($a>0){//si los años son diferentes
                //para el primer año
                $ano=$anoFin-$a;
                $chart=tipo_grafico($tipoGrafico);
                $dataSet = new XYDataSet();
                for($i=$mesIni;$i<=12;$i++){
                    $fini=$ano."/".$i."/01";
                    $ffin=$ano."/".$i."/31";
                    $s="select avg(m.valor) from Medida
m,Componente c,Nodo n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
n.id='$idusuario' and m.fecha between '$fini' and '$ffin' and m.tipo='$sensores[0]'";
                    $c=mysql_query($s,$conectar);
                    while ($g=mysql_fetch_array($c)){
                        $m=transforma_mes($i);
                        if ($g[0]==""){
                            $media=0;
                        }
                        else{
                            $media=$g[0];

```



```

    }
    $dataSet->addPoint(new Point($m,
$media));
    }
}
$chart->setDataSet($dataSet);
$chart->setTitle("Medidas de ".$sensores[0]." medias
de ".transforma_mes($mesIni)." a Diciembre de ".$ano);
$chart->render("/var/www/pfc/images/graficos/"
$sensores[0].$ano.$mesIni.".png");
echo '<p></p>';
//actualizo
$a--;
while($a>0){//para los del medio
    $ano=$anoFin-$a;
    $chart=tipo_grafico($tipoGrafico);
    $dataSet = new XYDataSet();
    for($i=1;$i<=12;$i++){
        $fini=$ano."/".$i."/01";
        $ffin=$ano."/".$i."/31";
        $s="select avg(m.valor) from Medida
m,Componente c,Nodo n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
n.id='$idusuario' and m.fecha between '$fini' and '$ffin' and m.tipo='$sensores[0]';
        $c=mysql_query($s,$conectar);
        while ($g=mysql_fetch_array($c)){
            $m=transforma_mes($i);
            if ($g[0]==""){
                $media=0;
            }
            else{
                $media=$g[0];
            }
            $dataSet->addPoint(new
Point($m, $media));
        }
    }
    $chart->setDataSet($dataSet);
}

```

```

                                $chart->setTitle("Medidas de ".$sensores[0]."
medias de Enero a Diciembre de ".$ano);
                                $chart-
>render("/var/www/pfc/images/graficos/".$sensores[0].$ano."0112.png");
                                echo '<p></p>';
                                $a--;
                                }
                                //para el ultimo año
                                $ano=$anoFin-$a;
                                $chart=tipo_grafico($tipoGrafico);
                                $dataSet = new XYDataSet();
                                for($i=1;$i<=$mesFin;$i++){
                                    $fini=$ano."/".$i."/01";
                                    $ffin=$ano."/".$i."/31";
                                    $s="select avg(m.valor) from Medida
m,Componente c,Nodo n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and
n.id='$idusuario' and m.fecha between '$fini' and '$ffin' and m.tipo='$sensores[0]'";
                                    $c=mysql_query($s,$conectar);
                                    while ($g=mysql_fetch_array($c)){
                                        $m=transforma_mes($i);
                                        if ($g[0]==""){
                                            $media=0;
                                        }
                                        else{
                                            $media=$g[0];
                                        }
                                        $dataSet->addPoint(new Point($m,
$media));
                                    }
                                }
                                $chart->setDataSet($dataSet);
                                $chart->setTitle("Medidas de ".$sensores[0]." medias
de Enero a ".$transforma_mes($mesFin)." de ".$ano);
                                $chart->render("/var/www/pfc/images/graficos/'.
$sensores[0].$ano.$mesFin.".png");
                                echo '<p></p>';
                                //actualizo

```

```

        $a--;
    }
    else { //son del mismo año
        $ano=$anoFin;
        $chart=tipo_grafico($tipoGrafico);
        $dataSet = new XYDataSet();
        echo "<strong><font color=\"red\">Maximo <font
color=\"blue\">Minimo <font color=\"orange\" >Media <font color=\"black\">de ".
$sensores[0]." entre los meses ".transforma_mes($mesIni)." y
".transforma_mes($mesFin)." de ".$ano."</strong>";
        echo "<br><br>";
        for($i=$mesIni;$i<=$mesFin;$i++){
            $fini=$ano."/".$i."/01";
            $ffin=$ano."/".$i."/31";
            $s="select avg(m.valor), max(m.valor),
min(m.valor) from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fini' and '$ffin' and
m.tipo='$sensores[0]'";

            $c=mysql_query($s,$conectar);
            while ($g=mysql_fetch_array($c)){
                $m=transforma_mes($i);
                if ($g[0]==""){
                    $media=0;
                    $mediaux="-";
                }
                else{
                    $media=$g[0];
                    $mediaux=$g[0];
                }
                if ($g[1]==""){
                    $max="-";
                }
                else{
                    $max=$g[1];
                }
                if ($g[2]==""){
                    $min="-";
                }
                else{

```

```

                                if ($g[2]<0){
                                    $minsel="select m.valor
from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha between '$fechaini' and
'$fechafin' and m.tipo='$sensores[0]' and m.valor<0 group by m.valor order by valor
desc";

                                $consmin=mysql_query($minsel,$conectar);

                                $datosmin=mysql_fetch_array($consmin);

                                $min=$datosmin[0];
                                }
                                else{
                                    $min=$g[2];
                                }
                                }
                                $dataSet->addPoint(new Point($m,
$media));
                                }
                                echo "<font
color=\"black\">".transforma_mes($i)." <font color=\"red\">".$max." <font
color=\"blue\">".$min." <font color=\"orange\">".$mediaux;
                                echo "<br>";
                                }
                                $chart->setDataSet($dataSet);
                                $chart->setTitle("Medidas de ".$sensores[0]." medias
de ".transforma_mes($mesIni)." a ".transforma_mes($mesFin)." de ".$ano);
                                $chart->render("/var/www/pfc/images/graficos/"
$sensores[0].$ano.$mesIni.$mesFin.".png");
                                echo '<p></p>';
                                }
                                echo "<p align=\"center\" ><font
color=\"black\">-----
-----</p><br>";
                                }
                                }
                                else {
                                    header("Location: graficas");

```

```

    }
}
else if (isset($opcion1)){
    //recojo parametros de fecha
    $diaIni=$_POST["diaini"];
    $mesIni=$_POST["mesini"];
    $anoIni=$_POST["añoini"];
    $diaFin=$_POST["diafin"];
    $mesFin=$_POST["mesfin"];
    $anoFin=$_POST["añoFin"];
    $tipoGrafico=$_POST["tipograf"];
    if ((! strstr($diaIni," ")) && (! strstr($mesIni," ")) && (! strstr($anoIni," "))
    && (! strstr($diaFin," ")) && (! strstr($mesFin," ")) && (! strstr($anoFin," "))) {
        $fechaini=$anoIni."/".$mesIni."/".$diaIni;
        $fechafin=$anoFin."/".$mesFin."/".$diaFin;
        echo "<big><strong><font color='red'>Maximo <font
color='blue'>Minimo <font color='orange'>Media <font color='black'>de ".
$sensores[0]." entre ".$fechaini." y ".$fechafin."</strong></big>";
        echo "<br><br>";
        $select="select m.tipo from Medida m,Componente c,Nodo n
where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaini' and '$fechafin' group by m.tipo";
        $consulta=mysql_query($select,$conectar);
        while($sensores=mysql_fetch_array($consulta)){
            $chart=tipo_grafico($tipoGrafico);
            $dataSet = new XYDataSet();
            $sfechas="select fecha from Medida m,Componente c,Nodo
n where m.id_comp=c.id_comp and c.id_nodo=n.id_nodo and n.id='$idusuario' and
m.fecha between '$fechaini' and '$fechafin' and m.tipo='$sensores[0]' group by fecha
order by fecha";
            $cfechas=mysql_query($sfechas,$conectar);
            while($fechas=mysql_fetch_array($cfechas)){
                $s="select avg(m.valor), max(m.valor), min(m.valor)
from Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha='$fechas[0]' and
m.tipo='$sensores[0]'";
                $c=mysql_query($s,$conectar);
                while ($g=mysql_fetch_array($c)){
                    if ($g[0]==""){

```

```

        $media=0;
        $mediaux="-";
    }
    else{
        $media=$g[0];
        $mediaux=$g[0];
    }
    if ($g[1]==""){
        $max="-";
    }
    else{
        $max=$g[1];
    }
    if ($g[2]==""){
        $min="-";
    }
    else{
        if ($g[2]<0){
            $minsel="select m.valor from
Medida m,Componente c,Nodo n where m.id_comp=c.id_comp and
c.id_nodo=n.id_nodo and n.id='$idusuario' and m.fecha='$fechas[0]' and
m.tipo='$sensores[0]' and m.valor<0 group by m.valor order by valor desc";
            $consmin=mysql_query($minsel,
$conectar);

            $datosmin=mysql_fetch_array($consmin);

            $min=$datosmin[0];
        }
        else{
            $min=$g[2];
        }
    }
    $dataSet->addPoint(new Point($fechas[0],
$media));
}
echo "<strong><font color=\"black\">Fecha:
</strong>".$fechas[0]." <font color=\"red\">".$max." <font color=\"blue\">".$min."
<font color=\"orange\">".$mediaux;
echo "<br>";

```

```

    }
    $chart->setDataSet($dataSet);
    $chart->setTitle("Medidas de ".$sensores[0]." medias de ".$fechaini." a ".$fechafin);
    $chart->render("/var/www/pfc/images/graficos/".$sensores[0].$fechas[0].".png");
    echo '<p></p>';
    echo "<strong><font color=\"00FF00\">Nota: </strong><font color=\"black\">En este apartado los días que tengan valores nulos (No hay datos) no se mostraran para garantizar la integridad del dibujo de la gráfica.";
    echo "<br><br>";
    echo "<p align=\"center\" ><font color=\"black\">-----
-----</p><br>";
    }
    }
    else {
        header("Location: graficas");
    }
}
?>

```

Estos son los script que hacen posible la utilización del panel de configuración. El primero de ellos es el rellena los campos con los datos del usuario.

```

<?php
    session_start();
    $user=$_SESSION["usuario"];
    //me conecto a la base de datos
    $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error conectando a mysql');
    $plant='<p><span style="font-size: large;"><strong><span style="color: #0000ff;">Datos personales</span></strong></span></p>
<p> </p>
<p> </p>
<form action="setconf" method="POST" enctype="multipart/form-data"><strong><span style="font-size: medium;">Nombre:</span></strong> <input type="text" name="nombre" value="##nombre##" /><br /> <span style="font-size:

```

```

medium;"><strong>Apellidos:</strong></span> <input type="text" name="apellidos"
value="###apellidos###" /><br /> <span style="font-size:
medium;"><strong>Usuario:</strong></span> <input type="text" name="usuario"
value="###usuario###" /><br /> <span style="font-size:
medium;"><strong>Contraseña:</strong></span> <input type="password" name="pass"
value="###pass###" /><br /> <span style="font-size: medium;"><strong>Rep.
Contraseña:</strong></span> <input type="password" name="reppass"
value="###pass###" /><br /> <span style="font-size:
medium;"><strong>Foto:</strong></span> <input id="foto" type="file" name="foto" /><br
/> <span style="font-size: medium;"><strong>Telefono:</strong></span> <input
type="text" name="telefono" value="###telefono###" /><br /> <span style="color:
#ffffff;">aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaa</span><input type="submit" name="boton" value="Modificar" /></form>;

//selecciono los datos del usuario de la sesion
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
//cojo la plantilla
$select="select nombre,apellidos,usuario,contrasena,telefono from Usuario
where usuario='$user'";
$consul=mysql_query($select,$conectar);
$u=mysql_fetch_array($consul);
$plantilla=str_replace("###nombre###",$u[0],$plant);
$plantilla=str_replace("###apellidos###",$u[1],$plantilla);
$plantilla=str_replace("###usuario###",$u[2],$plantilla);
$plantilla=str_replace("###pass###",$u[3],$plantilla);
$plantilla=str_replace("###telefono###",$u[4],$plantilla);
//y los imprimo
echo $plantilla;

?>

```

El segundo de ellos es el actualiza los cambios hechos por el usuario.

```

<?php
session_start();
$user=$_SESSION["usuario"];

```



```

//funcion con la cual joomla encripta las passwords
function pass_joomla($password){
    $salt = md5(mt_rand());
    $encrypt = md5($password.$salt);
    return $encrypt.'.'.$salt;
}
//recojo las variables
$nombre=mysql_real_escape_string($_POST["nombre"]);
$apellidos=mysql_real_escape_string($_POST["apellidos"]);
$usuario=mysql_real_escape_string($_POST["usuario"]);
$pass=mysql_real_escape_string($_POST["pass"]);
$reppass=mysql_real_escape_string($_POST["reppass"]);
/*Variable tipo fichero*/
$nombre_archivo = $_FILES["foto"]["name"];
$tipo_archivo = $_FILES["foto"]["type"];
$tamano_archivo = $_FILES["foto"]["size"];
$telefono=mysql_real_escape_string($_POST["telefono"]);
//me conecto a la base de datos
$conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de datos pfc');
if (stristr($pass,$reppass)){//si la contraseña y rep contraseña coinciden
    //actualizo todos los datos menos los de la foto
    $update="update Usuario set nombre='$nombre', apellidos='$apellidos',
usuario='$user', contrasena='$pass', telefono='$telefono' where usuario='$user'";
    mysql_query($update);
    if ($nombre_archivo!=""){
        if ((strpos($tipo_archivo,"jpeg")) || (strpos($tipo_archivo,"png"))){
            if ($tamano_archivo<70000){
                $path="/var/www/pfc/images/fotosResponsables/"
$user.".jpg";
                move_uploaded_file($_FILES["foto"]["tmp_name"],
$path);
                echo "<script language='JavaScript'>alert('Datos
personales modificados correctamente');</script>";
            }
        }
    }
}
else{

```

```

        echo "<script language='JavaScript'>alert('La foto no
puede ser superior a 70000 bytes');</script>";
    }
}
else {
    echo "<script language='JavaScript'>alert('La imagen debe
ser jpeg o png');</script>";
    $plant='<p><span style="font-size: large;"><strong><span
style="color: #0000ff;">Datos personales</span></strong></span></p>
<p> </p>
<p> </p>
<form action="setconf" method="POST" enctype="multipart/form-data"><strong><span
style="font-size: medium;">Nombre:</span></strong> <input type="text"
name="nombre" value="##nombre##" /><br /> <span style="font-size:
medium;"><strong>Apellidos:</strong></span> <input type="text" name="apellidos"
value="##apellidos##" /><br /> <span style="font-size:
medium;"><strong>Usuario:</strong></span> <input type="text" name="usuario"
value="##usuario##" /><br /> <span style="font-size:
medium;"><strong>Contraseña:</strong></span> <input type="password" name="pass"
value="##pass##" /><br /> <span style="font-size: medium;"><strong>Rep.
Contraseña:</strong></span> <input type="password" name="reppass"
value="##pass##" /><br /> <span style="font-size:
medium;"><strong>Foto:</strong></span> <input id="foto" type="file" name="foto" /><br
/> <span style="font-size: medium;"><strong>Telefono:</strong></span> <input
type="text" name="telefono" value="##telefono##" /><br /> <span style="color:
#ffffff;">aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaa</span><input type="submit" name="boton" value="Modificar" /></form>';
    //selecciono los datos del usuario de la sesion
    mysql_select_db(pfc,$conectar) or die ('Error conectado a la
base de datos pfc');

    //cojo la plantilla
    $select="select
nombre,apellidos,usuario,contrasena,telefono from Usuario where usuario='$user'";
    $consul=mysql_query($select,$conectar);
    $u=mysql_fetch_array($consul);
    $plantilla=str_replace("##nombre##",$u[0],$plant);
    $plantilla=str_replace("##apellidos##",$u[1],$plantilla);
    $plantilla=str_replace("##usuario##",$u[2],$plantilla);
    $plantilla=str_replace("##pass##",$u[3],$plantilla);

```

```

        $plantilla=str_replace("##telefono##",$u[4],$plantilla);
        //y los imprimo
        echo $plantilla;
    }
}
mysql_select_db(pfcjoomla,$conectar) or die ('Error conectado a la base
de datos pfcjoomla');
$p=pass_joomla($pass);
$update="update pfc_users set name='$nombre', username='$usuario',
password='$p' where username='$user'";
mysql_query($update);
header("Location: muro");
}
?>

```

Por último mostraré un api de funciones en php y el script que se encarga de descargar los fichero que el usuario tiene en la web.

Api funciones:

```

<?php
include "/var/www/pfc/libchart/libchart/classes/libchart.php";
function transforma_mes($i){
    if ($i==1){
        $m="Enero";
    }
    else if ($i==2){
        $m="Febrero";
    }
    else if ($i==3){
        $m="Marzo";
    }
    else if ($i==4){
        $m="Abril";
    }
    else if ($i==5){
        $m="Mayo";
    }
    else if ($i==6){

```

```

        $m="Junio";
    }
    else if ($i==7){
        $m="Julio";
    }
    else if ($i==8){
        $m="Agosto";
    }
    else if ($i==9){
        $m="Septiembre";
    }
    else if ($i==10){
        $m="Octubre";
    }
    else if ($i==11){
        $m="Noviembre";
    }
    else if ($i==12){
        $m="Diciembre";
    }
    return $m;
}

function tipo_grafico($tipo){
    if (strpos($tipo,"lineas")){
        $chart = new LineChart();
    }
    else if (strpos($tipo,"barras")){
        $chart = new VerticalBarChart();
    }
    else if (strpos($tipo,"sectores")){
        $chart = new PieChart();
    }
    return $chart;
}

function descargar($fichero,$path){
    header("Content-type: application/octet-stream");
    header("Content-Disposition: attachment; filename=\"\$fichero\".n");

```

```

        $fp=fopen("$path", "r");
        $b=fpassthru($fp);
        fclose($fp);
    }

```

```
?>
```

Script descargar fichero:

```

<?php
include "/var/www/pfc/phps/funciones.php";
$flagf=$_GET["flagf"];
if ($flagf!=""){
    //aqui procedo a descargar el fichero
    descargar("informe.doc","/var/www/pfc/ficheros/informes/informe.doc");
}
else{
    $user=$_GET["user"];
    $idel=$_GET["idfich"];
    //me conecto a la base de datos
    $conectar = mysql_connect("127.0.0.1","root","espectaculo") or die ('Error
conectando a mysql');
    mysql_select_db(pfc,$conectar) or die ('Error conectado a la base de
datos pfc');
    $sel="select nombre_fichero,tipo_fichero from Fichero where
id_fichero='$idel'";
    $nombrefichdel=mysql_fetch_array(mysql_query($sel,$conectar));
    $fichero=$nombrefichdel[0];
    $path="/var/www/pfc/ficheros/".$user.'/'.$fichero;
    descargar($fichero,$path);
    //header("Location: /index.php/misdocumentos");
}
?>

```

No he considerado oportuno mostrar el código fuente del programa auxiliar y de los script con los que se programan los dispositivos.

3.2 PROBLEMAS QUE HAN SURGIDO

Este va a ser el apartado donde se va a tratar todos los problemas que han surgido durante la construcción del proyecto.

Voy a dividir los problemas en tres grandes problemas, según las tres grandes partes de las que consta el proyecto, red de sensores, software auxiliar, página web.

Empezaré en primer lugar por la red de sensores. Este apartado, bajo mi punto de vista es clave. Hay que tomar la gran decisión de qué tecnología se va a usar para abordar el problema de la monitorización de las fincas.

A su vez también tienes que pensar muy bien a qué empresa vas a comprar la tecnología.

En cuanto a la tecnología hay que escoger una que, obviamente satisfaga los requisitos necesarios para que se pueda llevar a cabo el proyecto. Es condición indispensable.

Pero luego hay que tener en cuenta otros factores como la optimización de recursos, la eficiencia o la situación económica.

En cuanto a la optimización de recursos el principal factor a tener en cuenta sería el consumo de batería por parte de los dispositivos. Tema más que importante, ya que se pretende que la red de sensores trabaje de forma continua el máximo tiempo posible sin tener que recargar las baterías.

El tema de la eficiencia es importante respecto a los posibles fallos que se puedan producir al tomar las medidas o las posibles pérdidas a la hora de comunicar los distintos dispositivos. Por ello se debe elegir una tecnología que no provoque demasiados ruidos (fallos de los sensores al tomar medidas) y que la tecnología inalámbrica elegida proporcione una cobertura estable sin que se produzcan un número muy elevado de pérdidas de datos.

El tercer tema a valorar es tema del aspecto económico. Hay que tener una compensación entre la calidad y el precio del producto elegido. Ya que a mayor costo de la tecnología a usar mayor es el costo total del proyecto, que al final es lo que interesa al cliente final, que es el que paga.

Dicho esto, el precio de la tecnología considero que es secundario en relación con la eficiencia y por supuesto con el cumplimiento de objetivos, es decir, que por barata que sea una tecnología si no es satisfactoria para resolver el problema, habrá que obtener una que sí lo haga, aunque provoque un mayor desembolso económico.

A continuación comentaré los problemas y decisiones del software auxiliar empleado para la inserción de medidas en la base de datos.

He decir, que la mayoría de estos problemas que contaré a continuación son problemas surgidos a nivel de programación, es decir, que no son problemas de elección de recursos en los que hay que tener en cuenta diferentes aspectos.

El primero y principal es que el programa no se quedara colgado, ya que éste estará monitorizando un fichero cada cierto tiempo de manera indefinida. Se podría quedar colgado debido a una excepción, a datos incorrectos pasados por el usuario ...

Otro de los problemas es la utilización reiterativa de la base de datos, en cuanto a las conexiones, desconexiones, inserciones, consultas ... asegurarse que nunca falla, o que falla lo mínimo posible.

Como el programa monitoriza cada cierto tiempo el mismo fichero (el pasado por el usuario), este debe asegurarse de no insertar en la base datos las medidas que previamente ya hayan sido insertadas.

Una decisión importante es elegir la mejor manera de mostrar al usuario la correcta conexión y desconexión con el servidor de base de datos, la correcta inserción de una medida cada vez que se produzca o por el contrario si se ha producido algún error en cualquiera de los aspectos recién comentados notificarlo también.

En cuanto a programación pura, se puede decir que lo más complicado puede haber sido el trocear las medidas extraídas del fichero que se está monitorizando, ya que cada una de éstas vienen mezcladas con caracteres de cabecera del protocolo ZigBee, separadores elegidos a propósito por el proyectista ...

Otra elección importante es saber qué paradigma de programación se va a usar, estructurado, funcional, orientado a objetos ...

Una vez elegido el paradigma la elección recae sobre el lenguaje de programación en el que se va implementar el programa.

Por último, y aunque el usuario solo tenga que ejecutar una vez el programa proporcionar al usuario una interfaz sencilla e interactiva para la inserción de parámetros (IP, login, password de la base datos y el fichero a monitorizar).

En la tercer y última parte del proyecto, la página web, que es donde más problemas y más complicados me han surgido a la hora del diseño y de la implementación sobre todo.

La primera elección para codificación de la web fue la implementar “a mano” todo el código HTML (lenguaje gráfico para páginas web interpretado por un navegador), donde las restricciones de diseño se las pone uno mismo, o si en cambio usar un gestor de contenidos tales como Joomla, Drupal, Dreamwiewer ... , el cual te ayuda muchísimo en el diseño gráfico de la web pero, en cambio, tener que adecuarse (o estudiarlas para poder modificarlas) a las restricciones propias del gestor, de su base de datos propia ...

Al final con esta elección lo que se pretende es el usuario pueda disfrutar de una interfaz web que no le resulte demasiado cargante, y que sin mucho esfuerzo pueda visualizar de manera clara todos los elementos que le van saliendo en la pantalla.

El segundo problema viene dado por el carácter privado de la página. Éste es un proyecto privado para la Universidad pública de Navarra en la que sólo los responsables de las fincas de ésta tienen acceso. Para ello se ha de elegir un mecanismo de autenticación de usuario (login) y contraseña (password) que verifique al usuario.

Un problema derivado del anterior, es que al ser una web privada con usuarios, cada uno de estos, una vez logueados tiene la web personalizada con sus datos, dispositivos, medidas, datos personales, ficheros subidos ... Es importante elegir el método con el cual se van a identificar las consultas SQL, para la extracción de datos pedidos por las herramientas de la página con el usuario.

Se debe elegir cuál va a ser el lenguaje de programación dinámico, que se ejecutará en el servidor web (el que le va a dar la funcionalidad a la web), tal como Java, PHP, ASP, FLASH ...

Un aspecto a tener en cuenta, aunque en principio no parezca importante, es saber qué acciones tienen que ser enviadas al servidor para que se ejecuten allí y cuales pueden ser ejecutadas en el navegador del usuario (mediante JavaScript) para evitar en la medida de lo posible posibles retardos que al usuario se le hagan pesados.

Hay que saber que siempre que una acción es ejecutada en el servidor tiene más retardo que si es ejecutada en el cliente, ya que tiene que viajar al servidor, ser ejecutada y los resultados volver a viajar al navegador del usuario.

Los siguientes problemas que serán planteados son propios de la programación de la página web.

Uno de los aspectos fundamentales es el de sincronizar las medidas tomadas por los dispositivos de cada usuario con las todas alarmas creadas por ese usuario, de manera que cada medida que se vea afectada por cualquiera de las alarmas creadas deberá aparecer claramente diferenciada de las demás junto con el nombre la alarma que ha disparado.

Además el usuario deseaba que al pinchar en el nombre de la alarma que ha disparado la medida se refresque la pantalla la pantalla y justo debajo aparecen las características de la alarma seleccionada, en lugar de ir a la sección Muro donde están el listado de todas las alarmas.

Otro de los principales problemas que me han surgido es la representación gráfica de la finca del usuario donde coloca sus dispositivos.

El objetivo claro era que el usuario con simple click de ratón pudiera aparecer y desaparecer un dispositivo del mapa.

Hay funciones en lenguaje dinámico como PHP, que tienen efectos distintos si actúan sobre un gestor de contenidos que sobre un documento html normal. Un ejemplo es la función de descarga de documentos `fpassthru`. Hay que buscar alguna forma para que funcione perfectamente.

En el apartado de gráficas uno de los principales problemas es elegir las divisiones del intervalo elegido por el usuario para ser representado. Quiero decir que, por ejemplo, si el usuario elige un intervalo por días la división de los puntos de la gráfica está claro que es por días.

Al igual que si el usuario lo elige por meses la división será por meses.

La cosa es, que si es por años, la división será por igualmente por meses y de manera ordenada obviamente.

Al hilo del apartado de gráficas, también se tuvo que tener en cuenta que los meses en los que se divide el intervalo que no tuviesen medidas aparecen también representados en la gráfica.

Pero se comporta de distinta manera si la división del intervalo, en lugar de por meses, fuese por días. Si es por días, si alguno de estos no tienen medidas, no aparecerán representados en la gráfica. Esto es así para garantizar la integridad de la gráfica, ya que si el intervalo elegido por el usuario es bastante amplio (muchos días) y además se incluyen los días nulos quedaría un gráfico desproporcionado y bastante engorroso.

3.3 SOLUCIONES ADOPTADAS

En este punto se va a justificar las soluciones que he ha adoptado contra los problemas surgidos durante la construcción del proyecto.

Para, formar la red de sensores me he decidido por una red de dispositivos Waspote de la empresa Libelium.

Libelium diseña y fabrica tecnología de hardware para la implementación de redes sensoriales inalámbricas de forma que integradores de sistemas, ingenierías y consultorías puedan llevar a cabo soluciones fiables para Ciudades Inteligentes a usuarios finales, en el mínimo tiempo de salida al mercado.

Las principales líneas de producción y desarrollo son los dispositivos Waspote (los que se van a usar para este proyecto) y los router tecnológicos Meshilium.

Las principales características de los dispositivos Waspote son:

- Diseño de baterías para mínimo consumo (hasta 0,7 uA). Posee un sistema de alarmas e interrupciones para el cambio de estado.
- Soporte para casi todo tipo de sensores con su respectivas placas tales como gases (CO₂, CH₄, CO ...), temperatura, nivel líquido, peso, presión, humedad (ambiental y suelo), luminosidad, acelerómetro, radiación solar ...
- Acepta módulos de ZigBee, GPRS, Bluetooth, WiFi, GPS, tarjeta SD 2 GB ...
- Programable en C++ con un API Open Source (código libre). Totalmente compatible.

De Meshilium podemos decir que:

- Une cinco tecnologías en un mismo dispositivo, como ZigBee, 3G/GPRS, WiFi, Bluetooth, Ethernet.
- Corre en el una distribución del sistema operativo linux. Totalmente operativo.
- Capaz de detectar Smartphones bajo el sistema operativo iOS y Android.

- Cubierta resistente al agua, con conectores solares. [2]

A continuación voy mostrar detalladamente la tecnología inalámbrica ZigBee, que es la que he usado para montar la red entre los dispositivos Waspote, de Libelium.

Lo primero que voy a mostrar es un listado de los objetivos que se quieren explicar sobre ZigBee en este apartado:

- Dar a conocer en qué consiste y cómo operan las redes Zigbee.
- Conocer sus características, ventajas y desventajas de esta tecnología.
- Determinar los parámetros a tomar en cuenta para el uso de Zigbee.
- Identificar los métodos de seguridad y comunicación que tiene esta tecnología.
- Explicar la integración de Zigbee con las tecnologías inalámbricas emergentes según los aspectos de seguridad, costos, topología, estructura, tasa de transferencia y aplicaciones.
- Comparar las redes Zigbee con las diferentes tecnologías inalámbricas.

Lo primero que voy a realizar es una breve introducción para empezar a contar conceptos:

Las tecnologías inalámbricas han adoptado con el paso del tiempo una manera más sencilla y cómoda de utilizar toda clase de dispositivos con el fin de mejorar el confort y las comunicaciones en general.

Esta investigación aborda la tecnología inalámbrica ZigBee, basada en el estándar 802.15.4 que por su poca introducción al mercado no es muy conocida a pesar de que no es muy reciente.

ZigBee comunica una serie de dispositivos haciendo que trabajen más eficiente entre sí. Es un transmisor y un receptor que usa baja potencia para trabajar y tiene como objetivo las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías. Es ideal para conexiones con diversos tipos de topología, lo que a su vez lo hace más seguro, barato y que no haya ninguna dificultad a la hora de su construcción porque es muy sencilla.

ZigBee es la tecnología inalámbrica del futuro que no tiene competencia fuerte con las tecnologías existentes debido a que sus aplicaciones son de automatización de edificios, hogareñas e industriales, especialmente para aplicaciones con usos de sensores.

Resumen

ZigBee es un estándar de comunicaciones inalámbricas diseñado por la ZigBee Alliance. Es un conjunto estandarizado de soluciones que pueden ser implementadas por cualquier fabricante. ZigBee está basado en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (wireless personal area network, WPAN) y tiene como objetivo las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

ZigBee es un sistema ideal para redes domóticas, específicamente diseñado para reemplazar la proliferación de sensores/actuadores individuales. ZigBee fue creado para cubrir la necesidad del mercado de un sistema a bajo coste, un estándar para redes Wireless de pequeños paquetes de información, bajo consumo, seguro y fiable.

Historia

El nombre "ZigBee" se deriva de los patrones erráticos comunicativos que hacen muchas abejas entre las flores durante la recogida de polen. Esto es evocador de las redes invisibles de las conexiones existentes en un entorno totalmente inalámbrico.

ZigBee se ha desarrollado para satisfacer la creciente demanda de capacidad de red inalámbrica entre varios dispositivos de baja potencia. En la industria ZigBee se está utilizando para la próxima generación de fabricación automatizada, con pequeños transmisores en cada dispositivo, lo que permite la comunicación entre dispositivos a un ordenador central.

Para llevar a cabo este sistema, un grupo de trabajo llamado Alianza ZigBee (ZigBee Alliance) formado por varias industrias, sin ánimo de lucro, la mayoría de ellas fabricantes de semiconductores, está desarrollando el estándar.

La alianza de empresas está trabajando codo con codo con IEEE para asegurar una integración, completa y operativa. Esta alianza en la cuales destacan empresas como Invensys, Mitsubishi, Philips y Motorola trabajan para crear un sistema estándar de comunicaciones, vía radio y bidireccional, para usarlo dentro de dispositivos de automatización hogareña (domótica), de edificios (inmótica), control industrial, periféricos de PC y sensores médicos.

Los miembros de esta alianza justifican el desarrollo de este estándar para cubrir el vacío que se produce por debajo del Bluetooth.

Esta nueva aplicación, definida por la propia ZigBee Alliance como el nuevo estándar global para la automatización del hogar, permite que las aplicaciones domóticas desarrolladas por los fabricantes sean completamente ínter operables entre sí, garantizando así al cliente final fiabilidad, control, seguridad y comodidad.

Además la ZigBee Alliance también deja disponible para su acceso la ZigBee Cluster Library, ofreciendo de este modo a los ingenieros y demás integradores, deseosos de trabajar bajo este estándar mundial idóneo para los servicios domóticos, bloques de construcción para aplicaciones con necesidades bajo el denominador común de la automatización residencial, reduciendo de este modo las labores de desarrollo y permitiendo implementaciones más precisas.

Definición

ZigBee es el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radios digitales de bajo consumo, basada en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (wireless personal area network, WPAN).

Su objetivo son las aplicaciones para redes Wireless que requieren comunicaciones seguras y fiables con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

Estándar IEEE 802.15.4

IEEE 802.15.4 es un estándar que define el nivel físico y el control de acceso al medio de redes inalámbricas de área personal con tasas bajas de transmisión de datos (low-rate wireless personal area network, LR-WPAN). La actual revisión del estándar se aprobó en 2006. El grupo de trabajo IEEE 802.15 es el responsable de su desarrollo. También es la base sobre la que se define la especificación de ZigBee, cuyo propósito es ofrecer una solución completa para este tipo de redes construyendo los niveles superiores de la pila de protocolos que el estándar no cubre.

Características

- ZigBee, también conocido como "HomeRF Lite", es una tecnología inalámbrica con velocidades comprendidas entre 20 kB/s y 250 kB/s.
- Los rangos de alcance son de 10 m a 75 m.
- Puede usar las bandas libres ISM (6) de 2,4 GHz (Mundial), 868 MHz (Europa) y 915 MHz (EEUU).
- Una red ZigBee puede estar formada por hasta 255 nodos los cuales tienen la mayor parte del tiempo el transceiver ZigBee dormido con objeto de consumir menos que otras tecnologías inalámbricas.
- Un sensor equipado con un transceiver ZigBee pueda ser alimentado con dos pilas AA durante al menos 6 meses y hasta 2 años.
- La fabricación de un transmisor ZigBee consta de menos circuitos analógicos de los que se necesitan habitualmente.
- Diferentes tipos de topologías como estrella, punto a punto, malla, árbol.
- Acceso de canal mediante CSMA/CA (acceso múltiple por detección de portadora con evasión de colisiones).
- Escalabilidad de red -- Un mejor soporte para las redes más grandes, ofreciendo más opciones de gestión, flexibilidad y desempeño.

- Fragmentación -- Nueva capacidad para dividir mensajes más largos y permitir la interacción con otros protocolos y sistemas.
- Agilidad de frecuencia -- Redes cambian los canales en forma dinámica en caso que ocurran interferencias.
- Gestión automatizada de direcciones de dispositivos - El conjunto fue optimizado para grandes redes con gestión de red agregada y herramientas de configuración.
- Localización grupal -- Ofrece una optimización adicional de tráfico necesaria para las grandes redes.
- Puesta de servicio inalámbrico -- El conjunto fue mejorado con capacidades seguras para poner en marcha el servicio inalámbrico.
- Recolección centralizada de datos -- El conjunto fue sintonizado específicamente para optimizar el flujo de información en las grandes redes.

Ventajas

- Ideal para conexiones punto a punto y punto a multipunto. Con ello favorece una mayor diversidad de comunicación.
- Diseñado para el direccionamiento de información y el refrescamiento de la red.
- Opera en la banda libre de ISM 2.4 Ghz para conexiones inalámbricas.
- Óptimo para redes de baja tasa de transferencia de datos.
- Alojamiento de 16 bits a 64 bits de dirección extendida.
- Reduce tiempos de espera en el envío y recepción de paquetes.
- Detección de Energía (ED).

- Baja ciclo de trabajo - Proporciona larga duración de la batería.
- Soporte para múltiples topologías de red: Estática, dinámica, estrella y malla.
- Hasta 65.000 nodos en una red.
- 128-bit AES de cifrado. Provee conexiones seguras entre dispositivos.
- Son más baratos y de construcción más sencilla.

Desventajas

- La tasa de transferencia es muy baja.
- Solo manipula textos pequeños comparados con otras tecnologías.
- Zigbee trabaja de manera que no puede ser compatible con bluetooth en todos sus aspectos porque no llegan a tener las mismas tasas de transferencia, ni la misma capacidad de soporte para nodos.
- Tiene menor cobertura porque pertenece a redes inalámbricas de tipo WPAN.

Estructura

Siguiendo el estándar del modelo de referencia OSI (Open Systems Interconnection), en

el gráfico, aparece la estructura de la arquitectura en capas. Las primeras dos capas, la física y la de acceso al medio MAC, son definidas por el estándar IEEE 802.15.4.

Las capas superiores son definidas por la Alianza ZigBee y corresponden a las capas de red y de aplicación las cuales contienen los perfiles del uso, ajustes de la seguridad y la mensajería.

Los cometidos principales de la capa de red son permitir el correcto uso del subnivel MAC y ofrecer un interfaz adecuado para su uso por parte del nivel inmediatamente superior. Sus capacidades, incluyendo el ruteo, son las típicas de un nivel de red clásico.

Por una parte, la entidad de datos crea y gestiona las unidades de datos del nivel de red a partir del payload del nivel de aplicación y realiza el ruteo en base a la topología de la red en la que el dispositivo se encuentra.

Por otra parte, las funciones de control del nivel controlan la configuración de nuevos dispositivos y el establecimiento de nuevas redes; puede decidir si un dispositivo colindante pertenece a la red e identifica nuevos routers y vecinos. El control puede detectar así mismo la presencia de receptores, lo que posibilita la comunicación directa y la sincronización a nivel MAC.

La trama general de operaciones (GOF) es una capa que existe entre la de aplicaciones y el resto de capas. La GOF suele cubrir varios elementos que son comunes a todos los

dispositivos, como el subdireccionamiento, los modos de direccionamientos y la descripción de dispositivos, como el tipo de dispositivo, potencia, modos de dormir y coordinadores de cada uno.

Utilizando un modelo, la GOF especifica métodos, eventos, y formatos de datos que son utilizados para constituir comandos y las respuestas a los mismos.

La capa de aplicación es el más alto definido por la especificación y, por tanto, la interfaz efectiva entre el nodo ZigBee y sus usuarios.

En él se ubican la mayor parte de los componentes definidos por la especificación: tanto los objetos de dispositivo ZigBee (ZigBee device objects, ZDO) como sus procedimientos de control como los objetos de aplicación que se encuentran aquí.



Tipos de dispositivos

Se definen tres tipos distintos de dispositivo ZigBee según su papel en la red:

Coordinador ZigBee (ZigBee Coordinator, ZC): El tipo de dispositivo más completo. Debe existir uno por red. Sus funciones son las de encargarse de controlar la red y los caminos que deben seguir los dispositivos para conectarse entre ellos, requiere memoria y capacidad de computación.

Router ZigBee (ZigBee Router, ZR): Interconecta dispositivos separados en la topología de la red, además de ofrecer un nivel de aplicación para la ejecución de código de usuario.

Dispositivo final (ZigBee End Device, ZED): Posee la funcionalidad necesaria para comunicarse con su nodo padre (el coordinador o un router), pero no puede transmitir información destinada a otros dispositivos. De esta forma, este tipo de nodo puede estar dormido la mayor parte del tiempo, aumentando la vida media de sus baterías. Un ZED tiene requerimientos mínimos de memoria y es por tanto significativamente más barato.

Funcionalidad

Según su funcionalidad también puede clasificarse en:

Dispositivo de funcionalidad completa (FFD): También conocidos como nodo activo. Es capaz de recibir mensajes en formato 802.15.4. Gracias a la memoria adicional y a la capacidad de computar, puede funcionar como Coordinador o Router ZigBee, o puede ser usado en dispositivos de red que actúen de interfaces con los usuarios.

Dispositivo de funcionalidad reducida (RFD): También conocido como nodo pasivo. Tiene capacidad y funcionalidad limitadas con el objetivo de conseguir un bajo coste y una gran simplicidad. Básicamente, son los sensores/actuadores de la red.

Un nodo ZigBee (tanto activo como pasivo) reduce su consumo gracias a que puede permanecer dormido la mayor parte del tiempo (incluso muchos días seguidos). Cuando se requiere su uso, el nodo ZigBee es capaz de despertar en un tiempo ínfimo, para volverse a dormir cuando deje de ser requerido.

Topología

- Estrella: uno de los dispositivos tipo asume el rol de coordinador de red y es responsable de inicializar y mantener los dispositivos en la red. Todos los demás dispositivos zigbee, conocidos con el nombre de dispositivos finales, hablan directamente con el coordinador.
- Rejilla: el coordinador ZigBee es responsable de inicializar la red y de elegir los parámetros de la red, pero la red puede ser ampliada a través del uso de routers ZigBee.
- Punto a punto: Existe un solo coordinador. A diferencia con la topología estrella, cualquier dispositivo puede comunicarse con otro siempre y cuando estén en el mismo rango de alcance circundante.

Comunicación y descubrimiento de dispositivos

Para que los dispositivos que forman una aplicación puedan comunicarse, deben utilizar un protocolo de aplicación compartido. Estas convenciones se agrupan en perfiles. Las decisiones de asociación se deciden en base a la coincidencia entre identificadores de clusters de entrada y salida, que son únicos en el contexto de un perfil dado y se asocian a un flujo de datos de entrada o salida en un dispositivo; las tablas de asociaciones mantienen los pares de identificadores fuente y destino.

En base a la información disponible, el descubrimiento de dispositivos puede adecuarse utilizando varios métodos distintos. Si se conoce la dirección de red, se pide la dirección IEEE utilizando unicast. Sino es así, se pide por broadcast, y la dirección IEEE forma parte de la respuesta. Los dispositivos hoja (end devices) responden con la dirección propia solicitada, mientras que routers y coordinadores envían también las direcciones de todos los dispositivos asociados a ellos.

Este protocolo extendido permite indagar acerca de dispositivos dentro de una red y sus servicios ofrecidos a nodos externos a la misma. Los endpoints pueden informar acerca de estos servicios cuando el protocolo de descubrimiento dirige mensajes a ellos. También pueden utilizarse servicios de emparejamiento oferta-demanda.

Los identificadores de cluster favorecen la asociación entre entidades complementarias por medio de tablas de asociación, mantenidas en los coordinadores ZigBee ya que estas tablas siempre han de estar disponibles en una red (los coordinadores son, de entre todos los nodos, los que con mayor seguridad dispondrán de una alimentación continua). Los backups a estas tablas, de ser necesarios para la aplicación, han de realizarse en niveles superiores. Por otra parte, el establecimiento de asociaciones necesita que se haya formado un enlace de comunicación; tras ello, se decide si adjuntar un nuevo nodo a la red en base a la aplicación y las políticas de seguridad.

Nada más establecerse la asociación pueden iniciarse las comunicaciones. El direccionamiento directo utiliza la dirección de radio y el número de endpoint; por su parte, el indirecto necesita toda la información relevante (dirección, endpoint, cluster y atributo) y la envía al coordinador de la red, que mantiene esta información por él y traduce sus peticiones de comunicación. Este direccionamiento indirecto es especialmente útil para favorecer el uso de dispositivos muy sencillos y minimizar el almacenamiento interno necesario. Además de estos dos métodos, se puede hacer broadcast a todos los endpoints de un dispositivo, y direccionamiento de grupos para comunicarse con grupos de endpoints de uno o varios dispositivos distintos.

Seguridad

La seguridad de las transmisiones y de los datos son puntos clave en la tecnología ZigBee. ZigBee utiliza el modelo de seguridad de la subcapa MAC IEEE 802.15.4, la cual especifica 4 servicios de seguridad.

Control de accesos: El dispositivo mantiene una lista de los dispositivos comprobados en la red.

Datos Encriptados: Los cuales usan una encriptación con un código de 128 bits.

Integración de tramas: Protegen los datos de ser modificados por otros.

Secuencias de refresco: Comprueban que las tramas no han sido reemplazadas por otras.

El controlador de red comprueba estas tramas de refresco y su valor, para ver si son las esperadas.

Modelo básico de seguridad

Las claves son la base de la arquitectura de seguridad y, como tal, su protección es fundamental para la integridad del sistema.

Las claves nunca deben transportarse utilizando un canal inseguro, si bien existe una excepción momentánea que se da en la fase inicial de la unión de un dispositivo desconfigurado a una red. La red ZigBee debe tener particular cuidado, pues una red ad hoc puede ser accesible físicamente a cualquier dispositivo externo y el entorno de trabajo no se puede conocer de antemano. Las aplicaciones que se ejecutan en concurrencia utilizando el mismo transceptor deben, así mismo, confiar entre sí, ya que por motivos de coste no se asume la existencia de un cortafuegos entre las distintas entidades del nivel de aplicación.

Los distintos niveles definidos dentro de la pila de protocolos no están separados criptográficamente, por lo se necesitan políticas de acceso, que se asumen correctas en su diseño. Este modelo de confianza abierta (open trust) posibilita la compartición de claves disminuyendo el coste de forma significativa. No obstante, el nivel que genera una trama es siempre el responsable de su seguridad.

Todos los datos de las tramas del nivel de red han de estar cifradas, ya que podría haber dispositivos maliciosos, de forma que el tráfico no autorizado se previene de raíz. De nuevo, la excepción es la transmisión de la clave de red a un dispositivo nuevo, lo que dota a toda la red de un nivel de seguridad único.

También se posible utilizar criptografía en enlaces punto a punto.

Aplicaciones

Los protocolos ZigBee están definidos para su uso en aplicaciones embebidas con requerimientos muy bajos de transmisión de datos y consumo energético.

Se pretende su uso en aplicaciones de propósito general con características auto organizativas y bajo coste (redes en malla, en concreto).

Puede utilizarse para realizar control industrial, albergar sensores empotrados, recolectar datos médicos, ejercer labores de detección de humo o intrusos o domótica.

La red en su conjunto utilizará una cantidad muy pequeña de energía de forma que cada dispositivo individual pueda tener una autonomía de hasta 5 años antes de necesitar un recambio en su sistema de alimentación.



Comparación con otras Tecnologías Inalámbricas

Comparación de Tecnologías Inalámbricas			
	Wi-fi	Bluetooth	ZigBee
Bandas de Frecuencias	2.4GHz	2.4GHz	2.4GHz, 868 / 915 MHz
Tamaño de Pila	~ 1Mb	~ 1Mb	~ 20kb
Tasa de Transferencia	11Mbps	1Mbps	250kbps (2.4GHz) 40kbps (915MHz) 20kbps (868MHz)
Números de Canales	11 - - 14	79	16 (2.4GHz) 10 (915MHz) 1 (868MHz)
Tipos de Datos	Digital	Digital, Audio	Digital (Texto)
Rango de Nodos Internos	100m	10m - 100m	10m - 100m
Números de Dispositivos	32	8	255 / 65535
Requisitos de Alimentación	Media - Horas de Batería	Media - Días de Batería	Muy Baja - Años de Batería
Introducción al Mercado	Alta	Media	Baja
Arquitecturas	Estrella	Estrella	Estrella, Árbol, Punto a Punto y Malla
Mejores de Aplicaciones	Edificio con Internet Adentro	Computadoras y Teléfonos	Control de Bajo Costo y Monitoreo
Consumo de Potencia	400ma transmitiendo, 20ma en reposo	40ma transmitiendo, 0.2ma en reposo	30ma transmitiendo, 3ma en reposo
Precio	Costoso	Accesible	Bajo
Complejidad	Complejo	Complejo	Simple

[1]

Una vez acabado con las soluciones para la red de sensores, me voy a centrar en el software de inserción de medidas en la base de datos.

Para evitar el posible problema de un posible cuelgue ante un bucle, que en teoría nunca acaba (monitoriza el fichero cada cierto tiempo de manera indefinida) me he decidido por el empleo de un proceso concurrente que se encargue de hacer todas las operaciones del bucle. De manera que el programa principal simplemente hace de interfaz (entrada de datos, salida de mensajes).

En el tema de conexiones, desconexiones ... del servidor de base de datos he decidido capturar todas las posibles excepciones que pudieran saltar por separado, para ver posibles errores, problemas de sincronización ...

Para que el programa no inserte las mismas medidas cada vez que empieza de nuevo a monitorizar el fichero, lo que he hecho es realizar una consulta previa para ver si dicha medida ya ha sido previamente insertada. Si no es así, la inserto.

Para mostrar al usuario los mensajes de acierto y error sobre conexiones a base de datos, inserción de medidas ... he optado por pasar el objeto (JTextArea) por referencia al proceso concurrente que es el que se encarga de todo.

En el tema del análisis y preparación de las medidas extraídas del fichero, simplemente decir que con ayuda de criterios de separación (símbolos especiales de principio, fin, separación) y algún api se ha llegado a una buena solución para insertar las medidas correctamente.

En cuanto a la elección del paradigma de programación me decidí claramente por el paradigma orientado a objetos.

Se trata de aprovechar al máximo su encapsulamiento, la relación entre objetos y el resto de las muchísimas ventajas que te proporciona dicho paradigma.

En cuanto al lenguaje de programación me decidí claramente por Java. Es un lenguaje orientado a objetos, multiplataforma. Con esto quiero decir que lo puedo ejecutar en cualquier ordenador, bajo cualquier sistema operativo (de ordenador) o en un dispositivo Meshilium ... Además el API que tiene Java, no hace falta ni comentarios. Te ayuda a tener una programación bastante fácil y estructurada.

En cuanto a la interfaz del programa he usado la librería java.swing, java.awt para que el usuario pueda lanzar el programa de una manera sencilla y que no le cueste demasiado.

En este último apartado, voy a hablar de las soluciones adoptadas para los conflictos que he tenido con la implementación de la página web.

Sobre la decisión de implementar el código html de la web, o en cambio, optar por utilizar un gestor de contenidos web me he decidido por la segunda opción.

Concretamente el gestor de contenidos que he usado es el proyecto Open Source Joomla en su versión 1.7.

Esta decisión tiene sus ventajas pero también algún inconveniente. La principal ventaja es que el gestor de contenidos posee plantillas web ya hechas por lo que te olvidas de la mayor de la implementación html de la página. Ahorra mucho trabajo. Además es un buen elemento para la seguridad de la web, ya que el propio gestor es el que se encarga de escapar caracteres extraños para evitar SQL Injection, codifica las contraseñas con un sistema de encriptación propio ...

Pero entre los inconvenientes hay que decir que causa bastantes problemas la estructura que Joomla hace de su código y sus estructuras de datos cuando he pretendido realizar ciertas modificaciones en diseño y en el comportamiento. Un claro ejemplo de este tema es la sincronización de la base de datos del proyecto con la base de datos propia que utiliza Joomla.

Para la privatización de la web, he usado un módulo de Joomla basado en usuario y contraseña.

Para obtener el identificador de sesión para el resto de herramientas de la web modifiqué el php de Joomla que controla el dicho mecanismo, me guardé en una variable de sesión el login del usuario.

Luego con dicha variable de sesión, haciendo la pertinente consulta SQL se obtiene el resto de atributos del usuario necesarios para el resto de las herramientas de la web.

En la elección del lenguaje que se ejecuta en el servidor (proporciona el dinamismo a la página) me he decantado por PHP.

Me he decantado por PHP porque es un lenguaje orientado al tratamiento de cadenas de caracteres (extenso API de funciones), ya que en la programación web hay un único tipo que es el String.

Además posee un API de funciones para la seguridad web que es bastante aceptable.

Además php puede ser ejecutado tanto en servidores web corriendo tanto en Linux como en Windows.

Para las órdenes que se ejecutan en los navegadores, me decantado por el lenguaje JavaScript.

Prácticamente las órdenes que se ejecutan en JavaScript son la muestra de mensajes como respuesta para alguna acciones ejecutadas por los usuarios .

La solución a la sincronización de las medidas con las alarmas (de cada usuario) ha sido programada en un mismo script PHP. En un primer lugar se obtienen todas las alarmas del usuario. Luego para cada medida, se comprobará una a una si vulnera alguna de las alarmas y dependiendo de si es así o no se actuará conforme a las especificación de requisitos.

Otro tema era que si el usuario quiere consultar las características de alarma que ha disparado una medida haciendo click en el nombre de la alarma se le refresca la página de manera que no tenga que ir al apartado de Muro para ver dichas características. Pues bien, lo que hago es volver a realizar la tarea de la consulta y justo después imprimir la alarma con sus características.

Para las funciones PHP que no responden igual en un gestor de contenidos traspasé dicho código a un script PHP aparte.

El ejemplo en cuestión es la función `fpassthru`, para descargar un fichero del servidor web.

El tema de la representación de la finca y sus dispositivos lo solucioné con una tabla dividida en sectores pequeños con un enlace y unas variables de manera que al hacer click en ese enlace se enviaban dichas variables a un script php donde se actuaba en consecuencia (leer especificación de requisitos).

Para las divisiones del intervalo elegido por el usuario, en el apartado de gráficas, cuando la consulta sea por años o meses, será por meses. Mientras que si la consulta es por días la división será por días.

4.1 PRUEBAS UNITARIAS

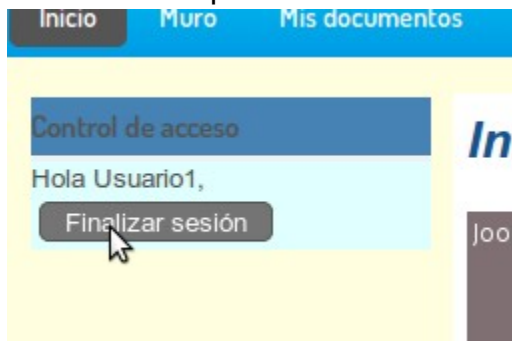
En será el apartado de pruebas del proyecto. En dicho apartado se realizarán las pruebas de unitarias para cada módulo PHP, que controla una determinada herramienta de la web y se evaluarán los resultados obtenidos.

Después se realizarán las pruebas de carga pertinentes para evaluar tiempos de espera, concurrencia ...

La primera prueba unitaria a evaluar será la autenticación de un usuario para entrar al sistema.

Para mostrar un ejemplo, el login es “user1” y la contraseña la misma, “user1”.

Al introducirlo podemos ver como se ha autenticado correctamente:



Ahora vamos a realizar las pruebas pertinentes a las herramientas de la barra de herramientas.



Empezaremos con el Muro.

El usuario (user1) al hacer click en la opción Muro le deberá salir sus datos personales junto con una foto.

A continuación un listado de las alarmas que hubiese creado con su respectivas características.

Datos responsable



Nombre:	Usuario1
Apellidos:	apellido1 apellido2
Email:	user1@pfc.com
Telefono:	666778898

Alarmas creadas

Nombre: Alarma 3

Tipo: Temperatura

Signo: < **Umbral:** 0

Descripcion: Temperaturas que bajan de 0 grados. Riesgo de heladas.

Solucion: Sistemas de botes de humo.

Nombre: Alarma 4





...

Se puede ver como, efectivamente, aparecen sus datos personales, su foto y un trozo del listado de las alarmas. Naturalmente continúa el listado, lo único que el pantallazo no daba para más.

En segundo lugar vamos a ver el comportamiento Mis documentos .
En está el usuario puede colgar los documentos que desee en el servidor web.
El ejemplo lo vamos a hacer con un documento llamado “repartidor”.

Antes de subirlo (se ve los documentos previamente alojados, si los hubiera).






Mis documentos

SoftonicDownloader_para_magix-web-d	 
enerotojunio2012.doc	 

Subir fichero:

Y después de subir el fichero el pantallazo resultante es:

Mis documentos

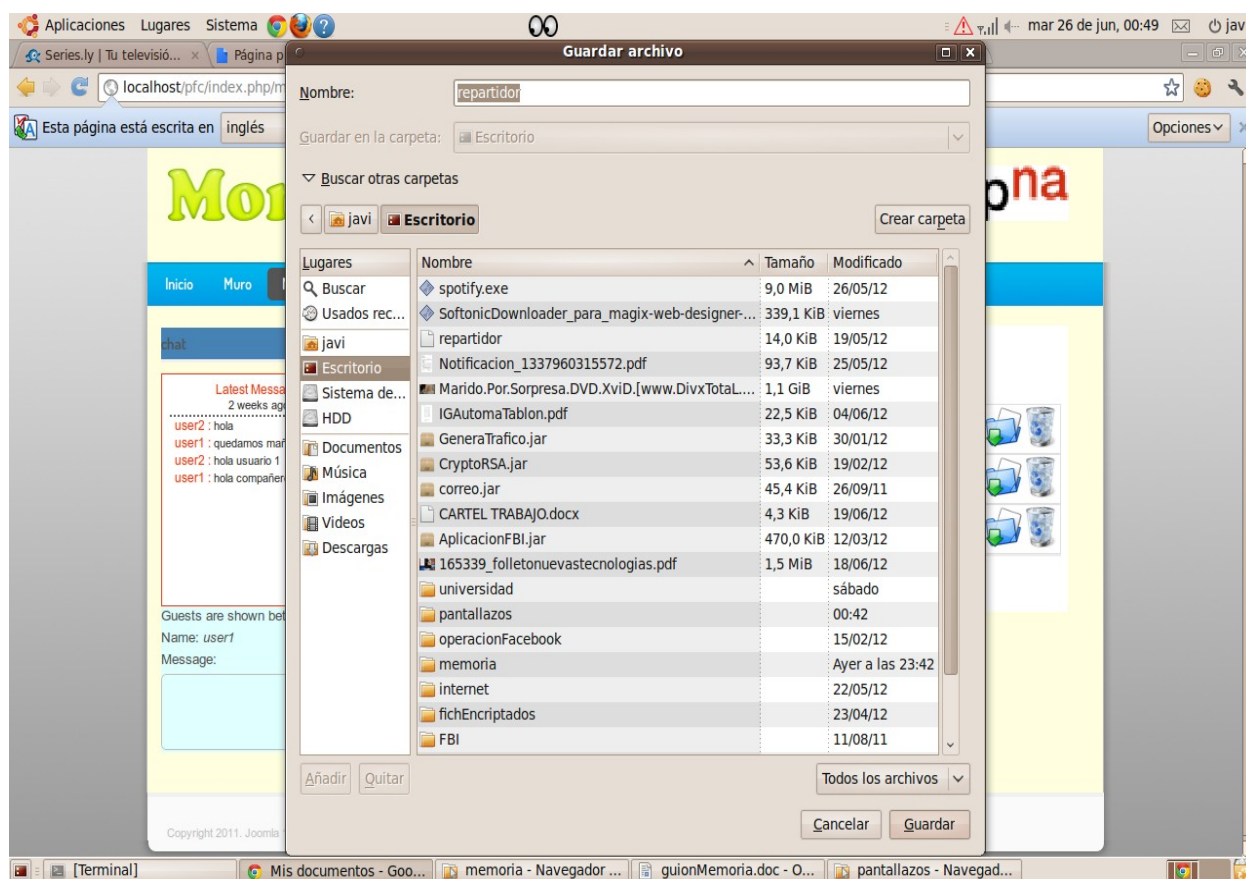
SoftonicDownloader_para_magix-web-d	 
enerotojunio2012.doc	 
repartidor	 

Subir fichero:

Efectivamente el fichero ha sido subido correctamente.

Los iconos de la derecha son los de descarga y borrado del fichero. Si pulsamos sobre la papelera del fichero repartido se eliminará dicho fichero volviendo al pantallazo de antes de la subida de dicho fichero.

En cambio si pulsamos sobre el botón de descargar aparecerá el cuadro para elegir donde descargarlo:

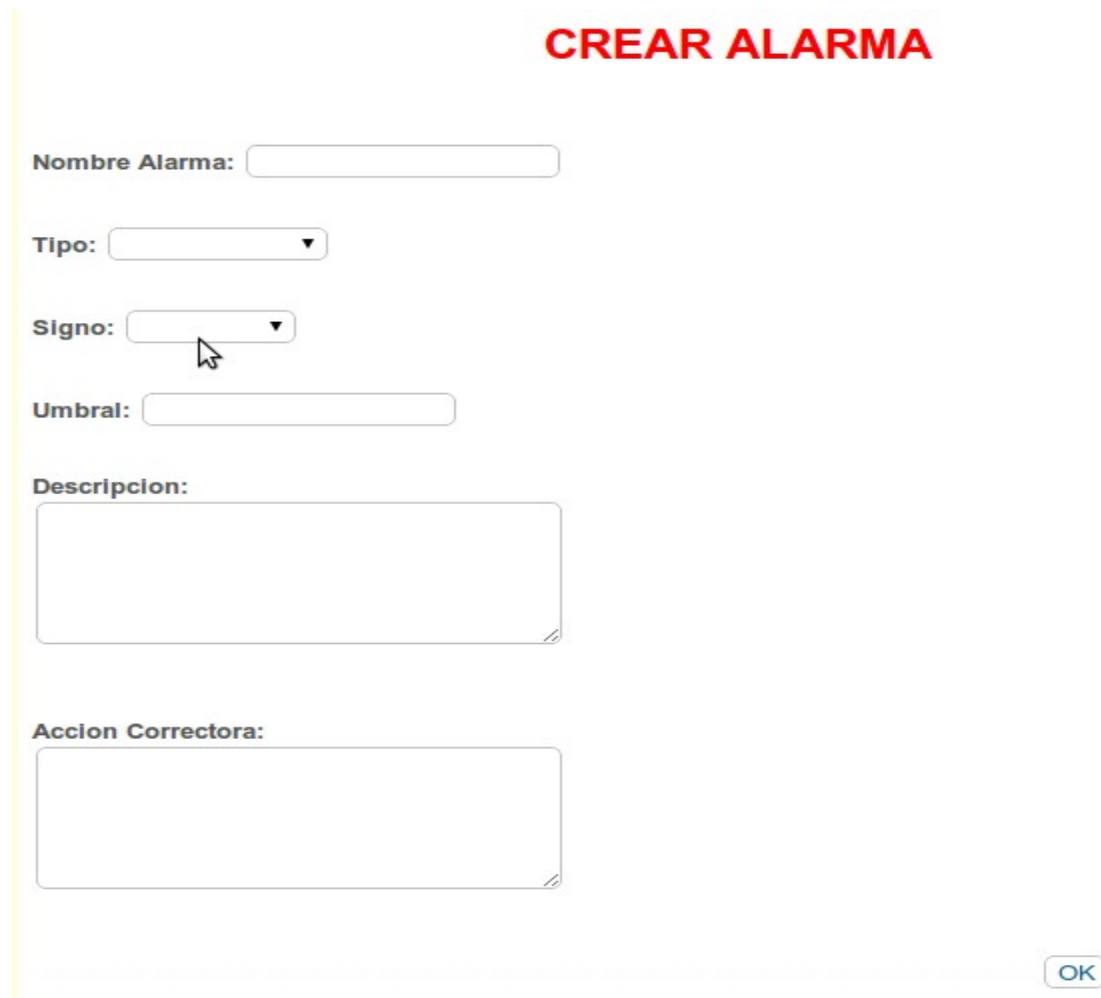


Efectivamente se ve el recuadro donde te dice donde guardar el fichero que el usuario ha querido descargar.

El siguiente apartado es el apartado de alarmas. Hay 3 opciones. Crear una nueva alarma, modificar una alarma o borrar una alarma.

Para la creación de una nueva alarma tenemos la siguiente interfaz. En ella se rellenan los datos y si son correctos se redirecciona a la página del muro donde aparece la alarma que se acaba de crear.

El pantallazo donde se muestra el muro demuestra la visualización de las alarmas creadas.



CREAR ALARMA

Nombre Alarma:

Tipo:

Signo:

Umbral:

Descripción:

Acción Correctora:

OK

Para la modificación y el borrado de alarmas se muestra un listado de las alarmas junto con el símbolo de modificar o borrar.

Si el usuario elige borrar desaparecerá dicha alarma y todo lo que conlleva (afectación medidas). En cambio si el usuario decide modificar aparecerá la interfaz anterior de creación de alarma con los campos rellenos por los datos de la alarma y actualizará dicha alarma con los cambios que haya hecho el usuario.

Interfaces modificación y borrado de alarmas.

MODIFICAR ALARMA

- Alarma 3: Temeperaturas que bajan de 0 grados. Riesgo de heladas.	
- Alarma 4: Exceso de humedad en la tierra. Posibles fallos de crecimiento	

BORRAR ALARMA

- Alarma 3: Temeperaturas que bajan de 0 grados. Riesgo de helada	
- Alarma 4: Exceso de humedad en la tierra. Posibles fallos de crecimiento	

El siguiente apartado a probar es el de consulta de medidas llamado Historial .

Se mostrará un ejemplo doble. Dependiendo de si el usuario quiere los resultados en la web o en un fichero local que se descargue previamente.

Como ejemplo el usuario desea hacer una consulta por días, concretamente el día 4 de marzo de 2012.

Consultas por días

En este apartado podrás realizar consultas por días. La aplicación mostrará todas las medidas que tus sensores hayan tomado en el intervalo de los dos días seleccionados (ambos inclusive).

Fecha Inicio: 4 ▼ Marzo ▼ 2012 ▼

Fecha Fin: 4 ▼ Marzo ▼ 2012 ▼

☐ Generar Informe

Consultar

Si no hace click en el checkbox Generar Informe los resultados se muestran en la misma web.

HISTORIAL DE CONSULTAS

1)	Tipo: CO2	Valor: 50	Fecha: 2012-03-04	Hora: 09:45	Nodo: 1200.	
2)	Tipo: CO2	Valor: 45	Fecha: 2012-03-04	Hora: 11:45	Nodo: 1200.	
3)	Tipo: Humedad	Valor: 19	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1300.	Alarma 4
4)	Tipo: Temperatura	Valor: -1	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1200.	Alarma 3

Datos globales: CO2

Medida máxima: 50 Medida mínima: 45 Media: 47.5

Datos globales: Humedad

Medida máxima: 19 Medida mínima: 19 Media: 19

Datos globales: Temperatura

Medida máxima: -1 Medida mínima: -1 Media: -1

Se visualiza su correcto funcionamiento.

Si el usuario hace click en el nombre de la alarma que ha disparado alguna de las medidas, por ejemplo la Alarma 4 (medida número 3) el resultado correcto sería el refresco de la pantalla con lo que había más las características de dicha alarma.

HISTORIAL DE CONSULTAS

1)	Tipo: CO2	Valor: 50	Fecha: 2012-03-04	Hora: 09:45	Nodo: 1200.	
2)	Tipo: CO2	Valor: 45	Fecha: 2012-03-04	Hora: 11:45	Nodo: 1200.	
3)	Tipo: Humedad	Valor: 19	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1300.	Alarma 4
4)	Tipo: Temperatura	Valor: -1	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1200.	Alarma 3

Datos globales: CO2

Medida máxima: 50 Medida mínima: 45 Media: 47.5

Datos globales: Humedad

Medida máxima: 19 Medida mínima: 19 Media: 19

Datos globales: Temperatura

Medida máxima: -1 Medida mínima: -1 Media: -1

Nombre: Alarma 4

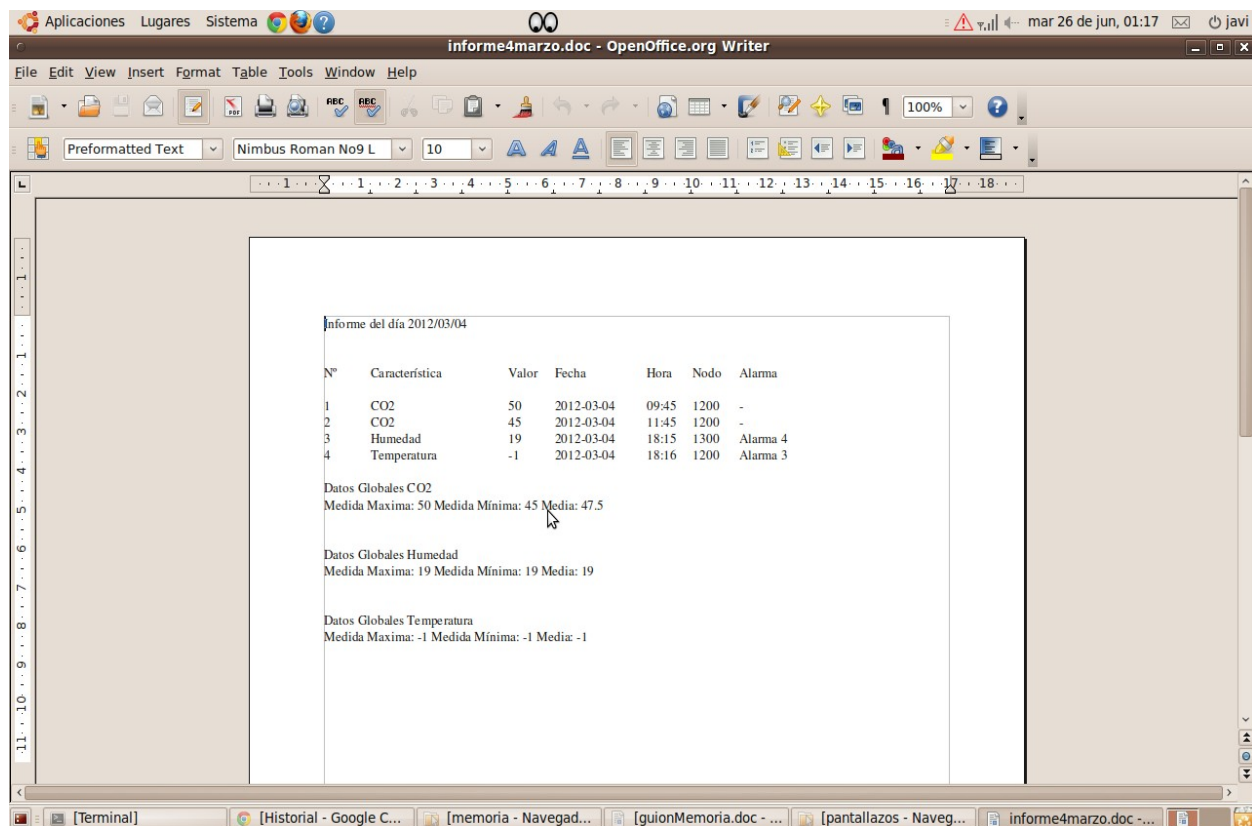
Tipo: Humedad

Signo: > **Umbral:** 1.25

Descripcion: Exceso de humedad en la tierra. Posibles fallos de crecimiento

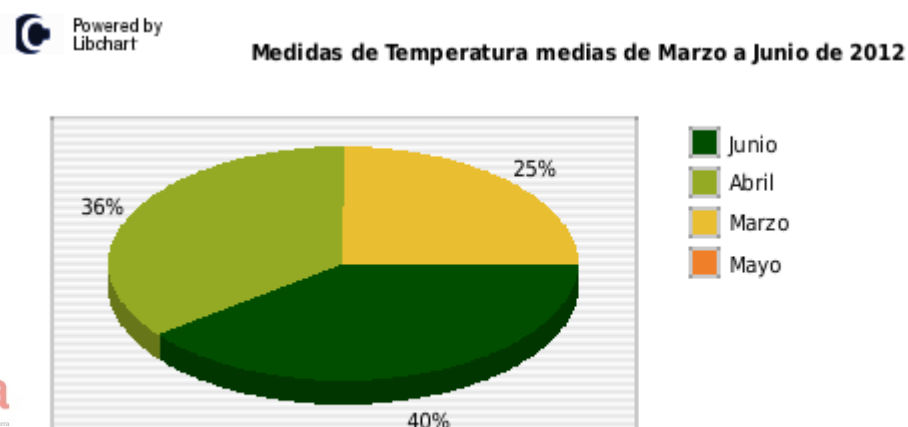
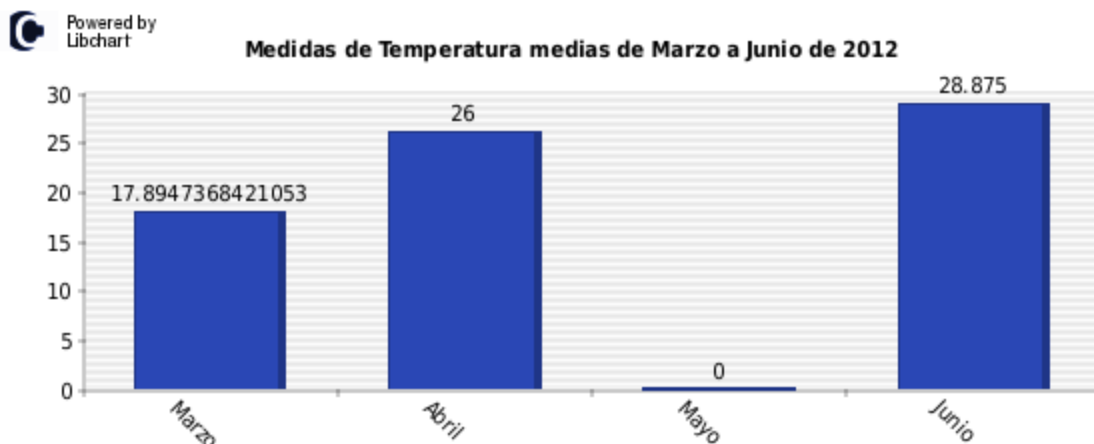
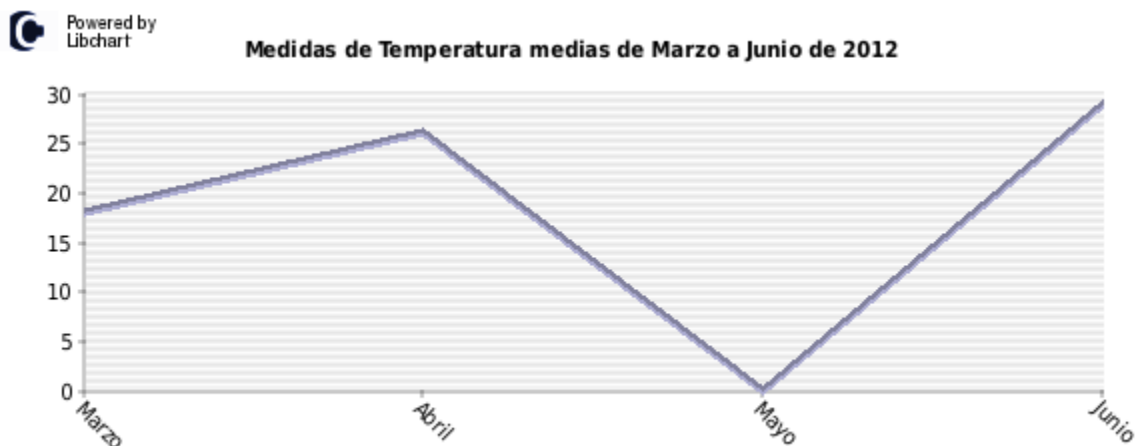
Solucion: No regar hasta que los niveles bajen.

En cambio si el usuario prefiere los datos en un documento en word el resultado sería:



A continuación mostraré el apartado de gráficas por un caso concreto, en los 3 tipo de gráficas presentes.

Concretamente mostraré una gráfica de los tipos de gráficos (líneas, barras y sectores) para Temperatura (obviamente el sistema proporciona las gráficas para el resto de sensores) entre los meses marzo a junio de 2012, ambos inclusive.



Como penúltimo módulo a mostrar voy a hablar de la herramienta Mi Finca .
Aquí el usuario puede hacer clic en una zona de una representación de su finca y colocar o quitar un dispositivo.

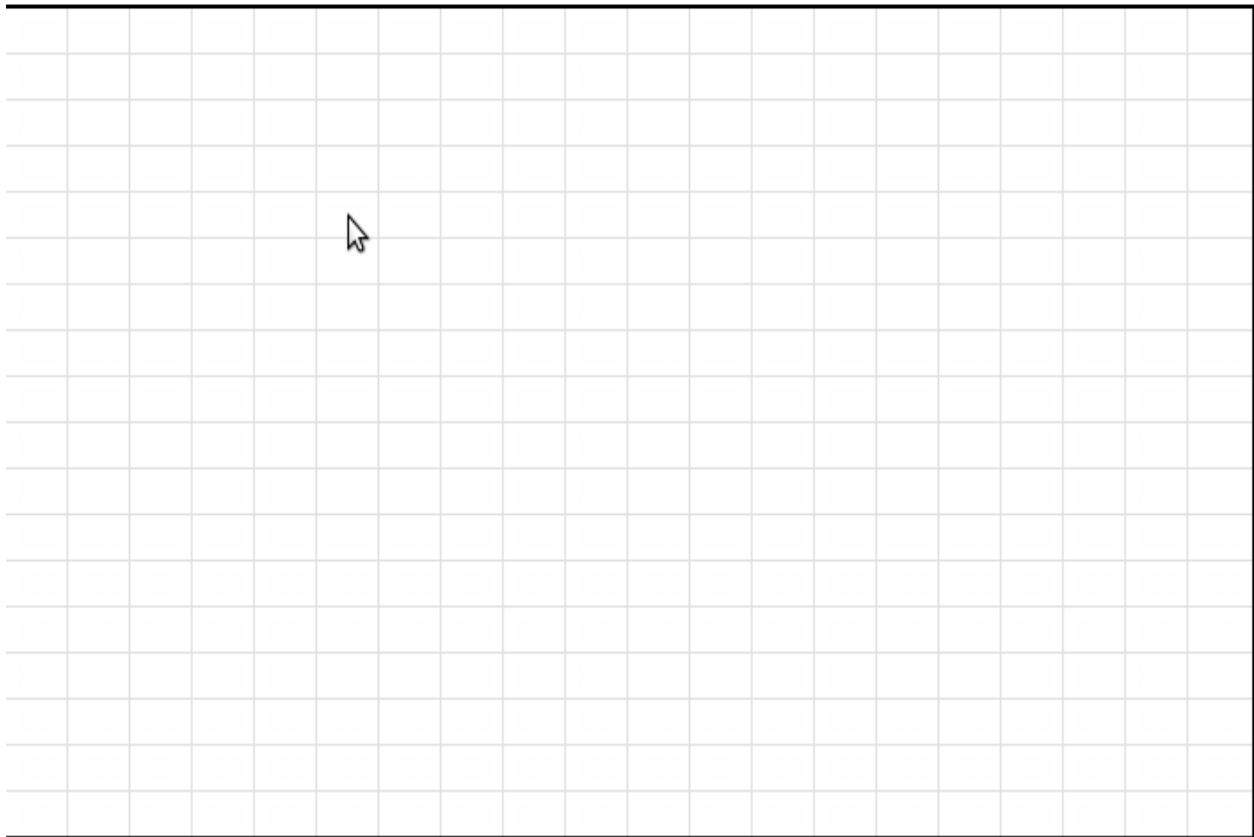
Antes de hacer ningún click. Finca limpia de dispositivos.



Haz click en la zona de la finca donde tengas el dispositivo waspmote. De esta manera te aparecerá un dibujito del dispositivo.

De igual manera para quitar el dispositivo del mapa de la finca haz click y el dispositivo desaparecerá.

Red User 1: Red que controla la finca del usuario 1



Después de hacer click donde esta señalando el ratón el mapa se refrescará y en esa zona debe aparecer un dispositivo WaspMote.

Análogamente si en la zona señalada por el usuario hay un dispositivo y hace click sobre él, dicho dispositivo será eliminado del mapa.

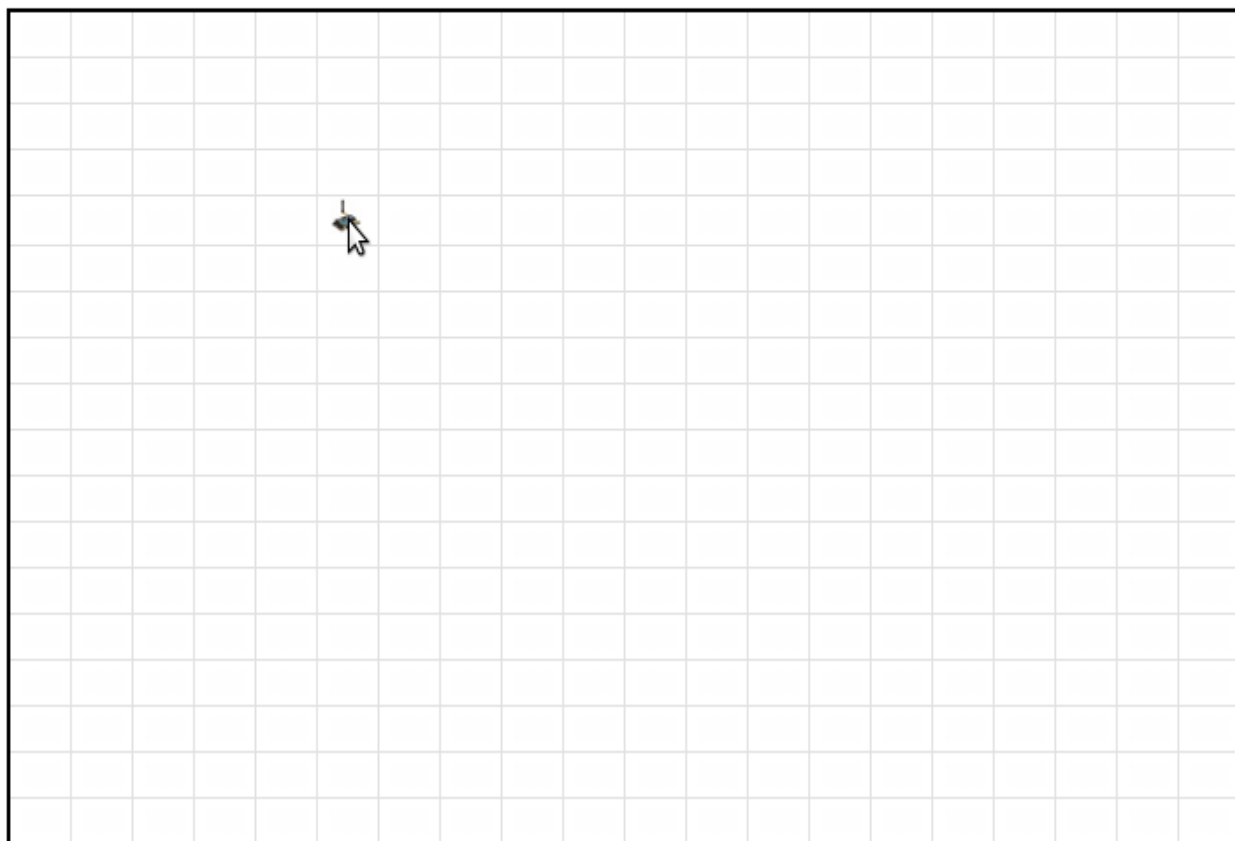
Ha aparecido un dispositivo en la zona vacía de la finca donde el usuario ha hecho click. Funcionamiento correcto.



Haz click en la zona de la finca donde tengas el dispositivo waspmote. De esta manera te aparecerá un dibujito del dispositivo.

De igual manera para quitar el dispositivo del mapa de la finca haz click y el dispositivo desaparecerá.

Red User 1: Red que controla la finca del usuario 1



El último apartado es el de configuración .

En él aparecen unos campos donde aparecen escritos los datos personales del usuario.



Datos personales

Nombre:

Apellidos:

Usuario:

Contraseña:

Rep. Contraseña:

Foto:

Telefono:

Una vez el usuario haya cambiado los datos pertinentes el sistema se redireccionará al Muro donde se comprobará como efectivamente los datos han sido correctamente modificados.

Con todo lo comentado, se han hecho las pruebas unitarias para cada uno de los módulos PHP que controlan las herramientas de la web.

4.2 PRUEBAS DE CARGA

A continuación comentaré varios ejemplos y observaciones que he visto al hacer pruebas de carga sobre el sistema web.

Una de las observaciones que he hecho es que, al ser una página web privada para los responsables de la fincas de la UPNA el número de peticiones de la web al servidor que se pudiesen dar simultáneamente no ocasionaría ningún problema (retardos, caídas del sistema ...).

Por lo tanto no sería necesario algoritmos ni infraestructura de redireccionamiento web (algoritmos de enrutamiento, proxies ...).

El acceso es instantáneo para cada uno de los usuarios.

En cuanto a las pruebas de carga en referencia al apartado del Historial (consulta de medidas), cuando el intervalo elegido por el usuario no excede en demasía el número de medidas el resultado es instantáneo. Si hubiese demasiadas medidas (consultas por años, por ejemplo) tarda un poco más, no demasiado.

La misma observación para el apartado de gráficas. Si la consulta es por días, y no hay muchos valores nulos (días sin tomar medidas) puede tardar algo más.

Lo que sí he notado, natural por otra parte, es que en el apartado de mis documentos, si el fichero a subir por el usuario demasiado grande, retardo ocasionado es proporcional al tamaño del fichero.

Los demás apartados como, Alarmas, Mi Finca, Muro y Configuración tienen unos resultados instantáneos.

5. PRESUPUESTO

En este apartado me dispondré a realizar un presupuesto del proyecto fin de carrera.

Antes debo decir, que he consultado diferentes fuentes para saber cómo valorar un proyecto informático una vez realizado. Y la verdad es que no me aclarado mucho ya que el precio por hora es muy variable. Incluso en algunos casos varía si la hora es dedicada al diseño (más cara) que a la implementación.

Suponiendo que valoro la hora dedicada al diseño en 40 € y la hora dedicada a la implementación 32 € voy a realizar un cálculo desglosado de lo que costaría el proyecto a partir de que el cliente me da la semántica describiendo el sistema informático que desea.

Empezaré con la etapa de análisis y diseño.

Comprende la etapa de la especificación de requisitos, diagramas pertinentes a varios niveles, pseudocódigo ...

Dura aproximadamente entre tres y cuatro semanas. Unos 25 días más o menos.

La cantidad de horas al día es bastante variable. Pero haciendo la media más o menos he hecho unas 4 horas.

3-4 semanas -> 25 días x 4 horas/días -> 100 horas * 40 €/hora = 4000 €

A continuación voy a valorar el periodo que me he pasado con la implementación a código fuente de los diseños hechos previamente.

Partiendo de que he trabajado 4 horas al día y estado unas 3-4 semanas (25 días) hasta que implementado todo. He usado hasta 7 lenguajes de programación distintos para llevar a cabo el proyecto (html, css, php, javascript, sql, java, c++).

3-4 semanas -> 25 días x 4 horas/días -> 100 horas * 32 €/hora = 3200 €

Por último entraré a valorar el tiempo que he tenido que emplear en realizar las pruebas pertinentes, para corregir los posibles fallos, y que el proyecto funcionase correctamente.

El precio de la hora para las pruebas es el mismo que el de implementación, es decir, 32 €

El tiempo que se estima de pruebas para un proyecto es muy relativo, no se puede predecir con anterioridad. Pero una vez evaluado y probado el sistema por cada módulo por separado puedo decir que el tiempo empleado es de dos semanas. 14 días.

Trabajando a 4 horas al día:

2 semanas -> 14 días x 4 horas/días -> 56 horas * 32 €/hora = 1792 €

El coste del mantenimiento del software en realidad, no existe. Ya que lo único que habría que estar pendiente es del cambio de baterías de los dispositivos Waspote y eso es cada muchísimo tiempo en teoría.

Pero si hay que añadir que me he pasado 3 semanas, 21 días, realizando la documentación pertinente del proyecto en el que incluye un manual de usuario. Cobrando la hora a la tarifa de implementación, 32 €, puedo decir que, al trabajar 3 horas diarias:

3 semanas -> 21 días x 3 horas/días -> 63 horas * 32 €/hora = 2016 €

Uniendo todo (análisis y diseño, implementación, pruebas y documentación) tenemos las siguientes cuentas.

Análisis y Diseño ->	4000 €
Implementación ->	3200 €
Pruebas ->	1792 €
Documentación ->	2016 €
+	

Proyecto ->	11008 €

El proyecto está valorado en 11008 €

6. CONCLUSIONES Y LÍNEAS FUTURAS

Se está llegando al final de la memoria y es momento de realizar las conclusiones. Hablando en líneas generales, para mí el proyecto fin de carrera ha sido la aplicación más larga y compleja que me ha tocado desarrollar hasta la fecha. Gracias a ello, me he dado cuenta de la importancia de las especificaciones, del análisis, gestión de recursos, decisiones a la hora de la implementación ...

Separando un poco el proyecto, en las tres partes citadas constantemente en la memoria, red de sensores, software de inserción auxiliar, y página web, voy a escribir las conclusiones por separado acorde a estas tres partes.

En cuanto a la red de sensores y todo lo que conlleva (dispositivos Waspote, tecnología ZigBee, programación en C++ ...) he encontrado la justificación suficiente por la cual elegí este proyecto fin de carrera. Me explico. Antes de elegir este proyecto miré unos cuantos más, pero lo que tenía claro es que quería hacer algo que usara además de un ordenador distintas plataformas electrónicas. Por eso, cuando vi que con este proyecto podía usar otros dispositivos electrónicos programables, como los dispositivos Waspote, encontré la suficiente motivación para realizar este proyecto. Gracias a ello también he aprendido el funcionamiento de una tecnología inalámbrica, como ZigBee, que no conocía y que, en mi opinión, se hará más pronto que tarde con una amplia cuota de mercado en el sector gracias a sus numerosas ventajas. De este punto, por último decir que también me motivaba bastante el hecho de empezar a aprender a programar en C++ ya que estaba bastante familiarizado con otros lenguajes orientados a objetos como Java y Visual Basic, pero me apetecía mucho aprender C++.

Las conclusiones el programa auxiliar no son muchas.

En esta parte proyecto he tenido que abordar la comunicación concurrente entre la base de datos, la red de sensores y el interfaz gráfico, lo que ha supuesto un reto importante.

Siempre me ha gustado combinar el máximo potencial de un lenguaje en un mismo programa.

Y en cuanto a la página web, tengo que decir, que estoy bastante orgulloso de la apariencia que tiene, pero sobre todo de la funcionalidad que he sido capaz de darle.

Antes del proyecto ya había programado alguna web con PHP pero tengo que decir que en el proyecto donde más he profundizado en el lenguaje, he utilizado más cosas y he aprendido más ha sido en éste.

En el tema de programación pura en PHP ha habido momentos, y alguno de ellos de larga duración que he tenido problemas y bloqueos, pero al final, con perseverancia siempre llegan los resultados deseados.

Pero lo que más destaco del desarrollo de la página web es la combinación de PHP con librerías externas y sobre todo el saber actuar para solucionar problemas a requisitos planteados en la semántica del proyecto.

Del primer punto me estoy refiriendo al sistema de gráficas. Tuve que emplear la librería de código abierto Libchart.

Al principio cuesta familiarizarte pero al final ves como poco a poco la cosa va dando resultado.

Con el segundo punto me refería concretamente al apartado de Mi Finca. Yo sabía que tenía que implementar un gráfico que interactuara de alguna manera con el usuario para la señalización de dispositivos Waspote en el gráfico. Al final diseñé un sistema de división del gráfico con enlaces y variables representando coordenadas útil y intuitivo.

Con este último término, intuitivo, es con el que me he querido basar para la construcción de la página web. Resumiendo, que al usuario no se le haga muy pesado ni tedioso el tener que realizar una tarea para que consiga lograr que el sistema haga lo que él desee.

Una vez acabado el apartado de conclusiones voy a hablar un poco sobre la posible ampliación de este proyecto y las posibles líneas futuras que abarca, que en mi opinión la informática y la agricultura son una simbiosis bastante productiva pero todavía en desarrollo.

A mí en concreto se me ocurren algunas ideas, que con más tiempo y recursos se podrían implementar mejorando bastante el proyecto.

La primera de ellas sería mostrar de alguna manera en el apartado de historial los posibles ruidos (fallos en la toma de medidas) que ocasionara un dispositivo en concreto.

La cosa es que, las medidas, aparte de por sensor, que como lo he hecho, aparezcan también ordenadas por dispositivo para saber si ha habido algún error en alguna medida y poder localizar dicho dispositivo inmediatamente. Todo esto ayudaría a evitar posibles fallos en temas de medidas medias, máximas y mínimas.

Otro avance que se me ha ocurrido es el de mejorar el gráfico del mapa que representa la localización de los sensores.

La mejora consiste en que, para cada tipo de sensor, aparezcan las zonas próximas a cada dispositivo coloreadas de un color representado el umbral del sensor elegido.

Un ejemplo sería si en distintas zonas de la finca la humedad en suelo que recoge cada sensor es distinta, puesto que ha podido llover y alguna zona ha quedado encharcada. En ese caso los dispositivos más cercanos a esa zona deberían colorearla de un color que represente un umbral más alto que el de los otros dispositivos.

Con este sistema se captaría mucha información acerca del estado de la finca de la manera más intuitiva posible que se me ocurre para el usuario.

Otra idea que me parece bastante buena sería la de poder juntar dos gráficas que representen el mismo sensor de un mismo intervalo de tiempo pero separados en el tiempo.

Por ejemplo, la temperatura del mes de enero por días, en año 2012 y 2011. Sería de gran ayuda para el usuario en lugar de estar mirando las dos gráficas por separado.

Hay alguna cosa más que se podría hacer, pero básicamente esto es lo que creo que convendría desarrollar en un futuro.

7. BIBLIOGRAFÍA

[1] Documento de la Universidad Nacional de Ingeniería de Nicaragua, sobre la tecnología inalámbrica ZigBee, por Carlos Alberto Ortega, Deyanira del Socorro y Eduardo Úbeda.

[2] Documentación sobre los dispositivos inalámbricos de la empresa Libelium <http://www.libelium.com/products/>

8. ANEXO I. MANUAL DE USUARIO

En las siguientes líneas documentaré esta memoria con el manual de usuario relativo al funcionamiento de la página web.

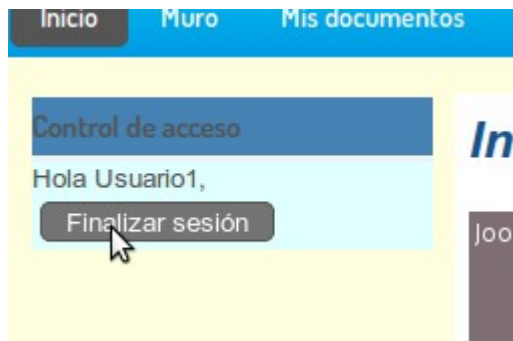
En primer lugar voy a realizar un pantallazo del índice de la página web para visualizar los principales elementos, sistema de autenticación, barra de herramientas, galería fotográfica ...



Al ser una página de carácter privado, naturalmente posee un sistema de autenticación basado en un formulario web de login y password.

Para visualizar un ejemplo, hay un usuario creado llamado "user1" con contraseña también "user1" (con este usuario es con el que haré todo el manual de usuario).

Si introducimos el login y password correcto el sistema nos dejará paso.



En cambio si los datos no son correctos no podremos usar ninguna de las herramientas de la web.

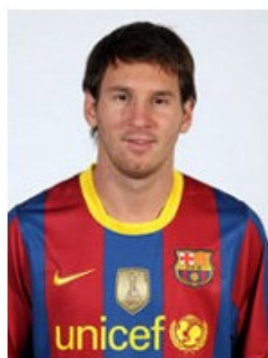


Una vez que estamos correctamente autenticados voy a mostrar cada una de las funciones de los items de la barra de herramientas.



Empezaré por la herramienta **Muro**. Aquí el usuario podrá visualizar sus datos personales (nombre, apellidos, teléfono ...) junto con una foto suya. A continuación se le mostrará un listado con todas las alarmas creadas por él.

Datos responsable



Nombre:	Usuario1
Apellidos:	apellido1 apellido2
Email:	user1@pfc.com
Telefono:	666778898

Alarmas creadas

Nombre: Alarma 3

Tipo: Temperatura

Signo: < **Umbral:** 0

Descripcion: Temperaturas que bajan de 0 grados. Riesgo de heladas.

Solucion: Sistemas de botes de humo.

Nombre: Alarma 4

El listado de las alarmas obviamente se ve que continúa. Pero el pantallazo lo trunca.

La segunda herramienta que vamos a demostrar es la de **Mis documentos**. En ella los usuarios podrán subir, descargar y borrar los ficheros que ellos deseen. Un sistema de alojamiento web sencillo para que el usuario pueda tener unido la mayor parte de su trabajo.

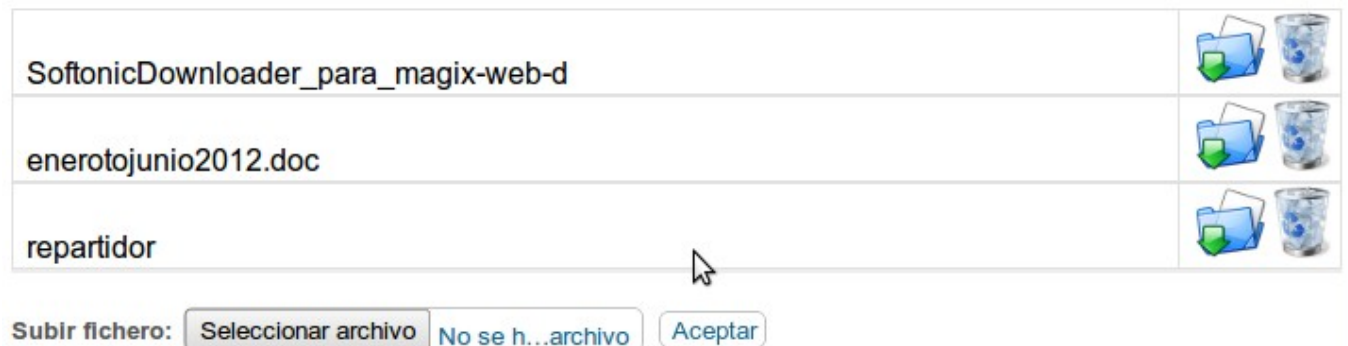
Se va a ver un ejemplo en el que se verán un par de pantallazos. El primero se visualizarán los ficheros que tuviese el usuario alojados en la web (si es que tuviese alguno) antes de subir un fichero llamado “repartidor”.

Mis documentos



A continuación, después de subir el fichero “repartidor”.

Mis documentos



Se puede observar como para cada uno de los fichero alojados existe un icono para descargarlo y otro para borrarlo. Con un simple click sobre cualquiera de ellos hace la acción correspondiente.

A continuación voy a mostrar la utilidad **Alarmas**. Aquí los usuarios podrán crear, modificar y borrar alarmas para llevar un control de posibles medidas críticas que se pudiesen dar en la finca.

Al hacer click sobre dicha pestaña aparecerá un menú sobre la acción a realizar:



Si pulsas sobre crear alarma te aparecerá un formulario web con los datos necesarios para crear una nueva alarma. No puedes tener dos alarmas que se llamen igual para un mismo usuario. Y todos los campos son obligatorios. Si no es así, el sistema te notificará el error pertinente y abortará la operación. En cambio, si todo es correcto creará una nueva alarma. Dicha alarma, obviamente afectará a las medidas tomadas por los sensores, y te redireccionara al Muro, donde se visualizarán todas las características de la nueva alarma (pantallazo visto antes).

Interfaz crear alarma:

CREAR ALARMA

Nombre Alarma:

Tipo:

Signo:

Umbral:

Descripcion:

Accion Correctora:

OK

En cambio si pulsas en Modificar Alarma, te aparecerá un listado con el nombre de cada alarma, su descripción y un icono de modificar.

MODIFICAR ALARMA

- Alarma 3: Temeperaturas que bajan de 0 grados. Riesgo de heladas.	
- Alarma 4: Exceso de humedad en la tierra. Posibles fallos de crecimiento	

Si le das al icono de cualquiera de ellas te aparecerá el formulario web de antes (Crea Alarma) pero con los campos rellenos con los datos de la Alarma. Una vez que el usuario da a modificar vuelve al listado de modificado.

En cambio si pulsas en Borrar Alarma, te aparecerá un listado con el nombre de cada alarma, su descripción y un icono de borrar.

BORRAR ALARMA

- Alarma 3: Temeperaturas que bajan de 0 grados. Riesgo de helada	
- Alarma 4: Exceso de humedad en la tierra. Posibles fallos de crecimiento	

Si le das al icono de borrar de cualquiera de ellas la alarma en cuestión, de manera que las medidas afectadas por dicha alarma ya no se verán afectadas.

El siguiente paso a explicar es el de **Historial**. Aquí el usuario podrá consultar las medias tomadas por la red de sensores, por años, meses o días. El resultado de la consulta se puede obtener por web o en fichero word que previamente será descargado por el usuario.

Voy a realizar un ejemplo en el que se quiere saber las medidas tomadas el 4 de marzo de 2012.

Consultas por días

En este apartado podrás realizar consultas por días. La aplicación mostrará todas las medidas que tus sensores hayan tomado en el intervalo de los dos días seleccionados (ambos inclusive).

Fecha Inicio: 4 Marzo 2012

Fecha Fin: 4 Marzo 2012

El resultado por web sería:

HISTORIAL DE CONSULTAS

1)	Tipo: CO2	Valor: 50	Fecha: 2012-03-04	Hora: 09:45	Nodo: 1200.	
2)	Tipo: CO2	Valor: 45	Fecha: 2012-03-04	Hora: 11:45	Nodo: 1200.	
3)	Tipo: Humedad	Valor: 19	Fecha: 2012-03-04	Hora: 18:15	Nodo: 1300.	Alarma 4
4)	Tipo: Temperatura	Valor: -1	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1200.	Alarma 3

Datos globales: CO2

Medida máxima: 50 Medida mínima: 45 Media: 47.5

Datos globales: Humedad

Medida máxima: 19 Medida mínima: 19 Media: 19

Datos globales: Temperatura

Medida máxima: -1 Medida mínima: -1 Media: -1

Como se puede observar las medidas afectadas por alguna alarmas aparecen en un formato distinto acorde a la especificación de requisitos.

Si el usuario pincha sobre el nombre de la alarma que disparado la medida la pantalla se refrescará añadiendo las características de esa alarma.

Por ejemplo si el usuario pincha en la alarma 4 el resultado es el siguiente:

HISTORIAL DE CONSULTAS

1)	Tipo: CO2	Valor: 50	Fecha: 2012-03-04	Hora: 09:45	Nodo: 1200.	
2)	Tipo: CO2	Valor: 45	Fecha: 2012-03-04	Hora: 11:45	Nodo: 1200.	
3)	Tipo: Humedad	Valor: 19	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1300.	Alarma 4
4)	Tipo: Temperatura	Valor: -1	Fecha: 2012-03-04	Hora: 18:16	Nodo: 1200.	Alarma 3

Datos globales: CO2

Medida máxima: 50 Medida mínima: 45 Media: 47.5

Datos globales: Humedad

Medida máxima: 19 Medida mínima: 19 Media: 19

Datos globales: Temperatura

Medida máxima: -1 Medida mínima: -1 Media: -1

Nombre: Alarma 4

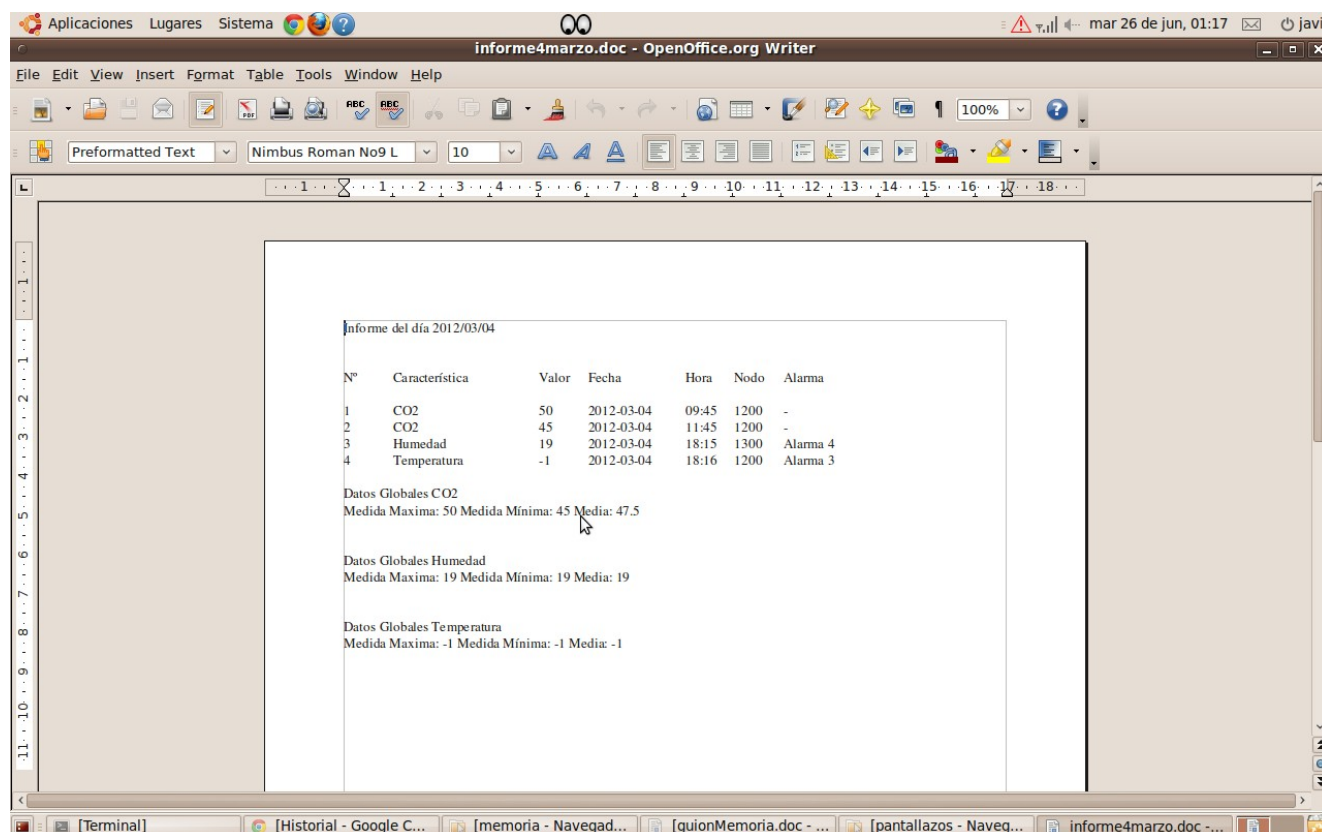
Tipo: Humedad

Signo: > **Umbral:** 1.25

Descripcion: Exceso de humedad en la tierra. Posibles fallos de crecimiento

Solucion: No regar hasta que los niveles bajen.

En cambio si el usuario desea descargar la consulta de las medidas en un documento de word debe pinchar el checkbox “Generar Informe” y después hacer click en consultar.



A continuación voy a explicar el apartado de **Gráficas**. Al igual que el historial el usuario puede elegir un intervalo por años, meses o días. La división del intervalo cuando es por años o meses es por meses. Sin embargo la división del intervalo es por días cuando la consulta es por días. La representación gráfica representa las medias de las divisiones del intervalo. Pero justo antes del gráfico también se muestran las máximas y mínimas junto con la media. Hay tres tipos de gráfico, líneas, barras, sectores.

Elección consulta gráfica:

Gráficas por meses

Descripción de los meses

Mes Inicio: Marzo 2012

Mes Fin: Junio 2012

☒ Líneas XY

☐ Barras

☐ Sectores

[Consultar](#)

Voy a mostrar los resultado de temperatura en este caso. Pero saldría para cada tipo de medida.

Los datos de arriba de la gráfica sólo los voy a mostrar una vez

Maximo Minimo Media de Temperatura entre los meses Marzo y Junio de 2012

Marzo 25 -5 17.8947368421053

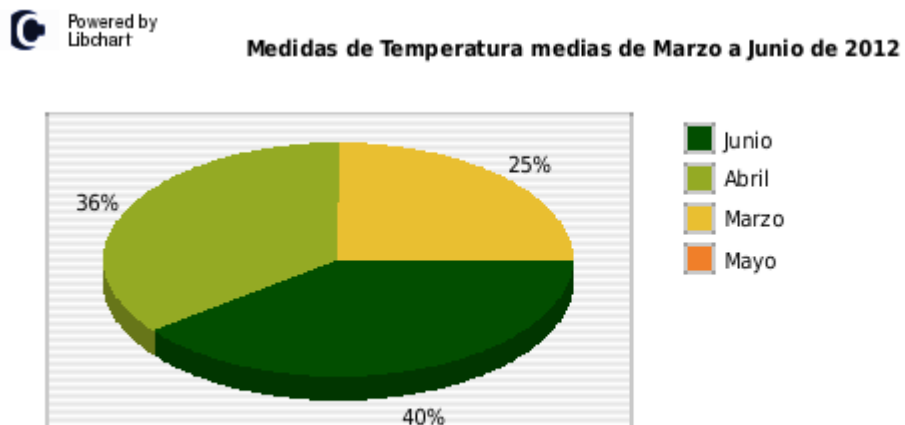
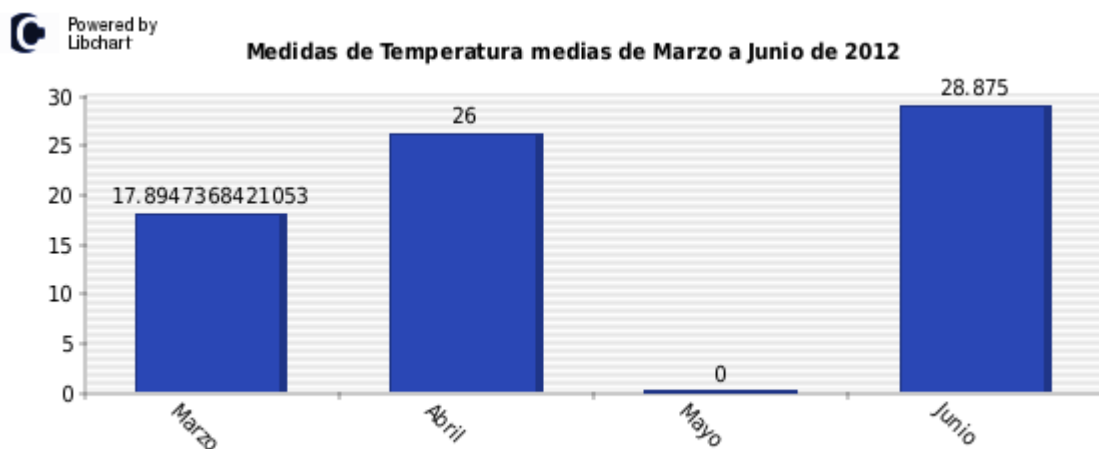
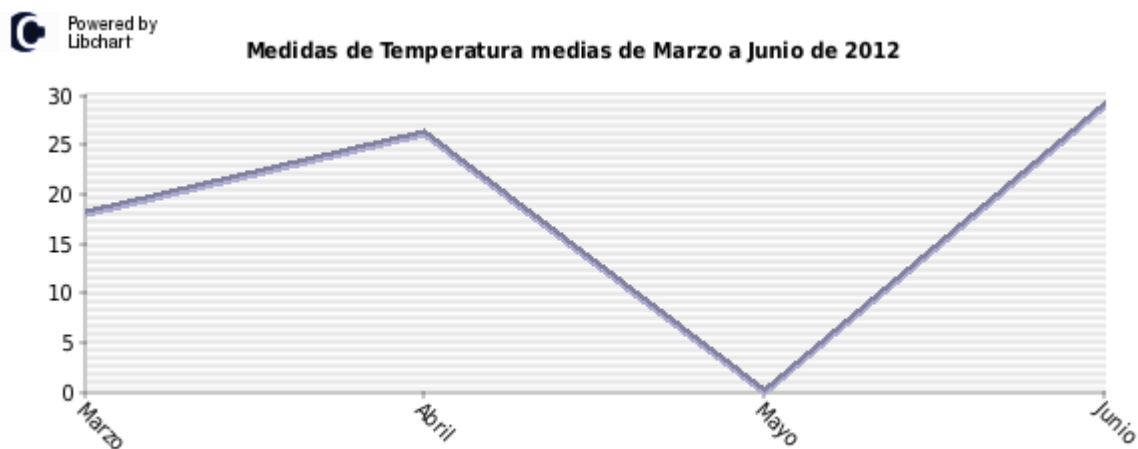
Abril 28 25 26

Mayo - - -

Junio 30 28 28.875

—

Gráfica de líneas, barras y sectores.



El siguiente apartado que voy a mostrar es el de **Mi Finca**.

Se trata de una representación gráfica de la finca del usuario donde éste podrá poner y quitar los dispositivos con un simple click de ratón

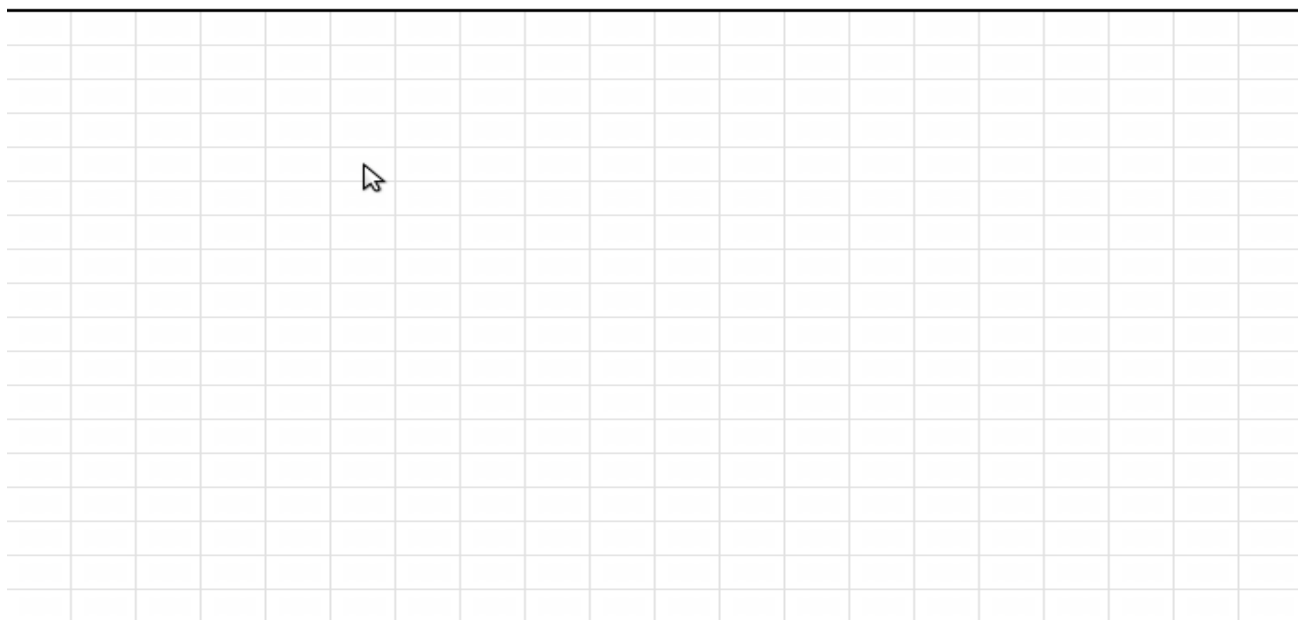
Representación gráfica de la finca “limpia” de sensores.



Haz click en la zona de la finca donde tengas el dispositivo waspmote. De esta manera te aparecerá un dibujito del dispositivo.

De igual manera para quitar el dispositivo del mapa de la finca haz click y el dispositivo desaparecerá.

Red User 1: Red que controla la finca del usuario 1



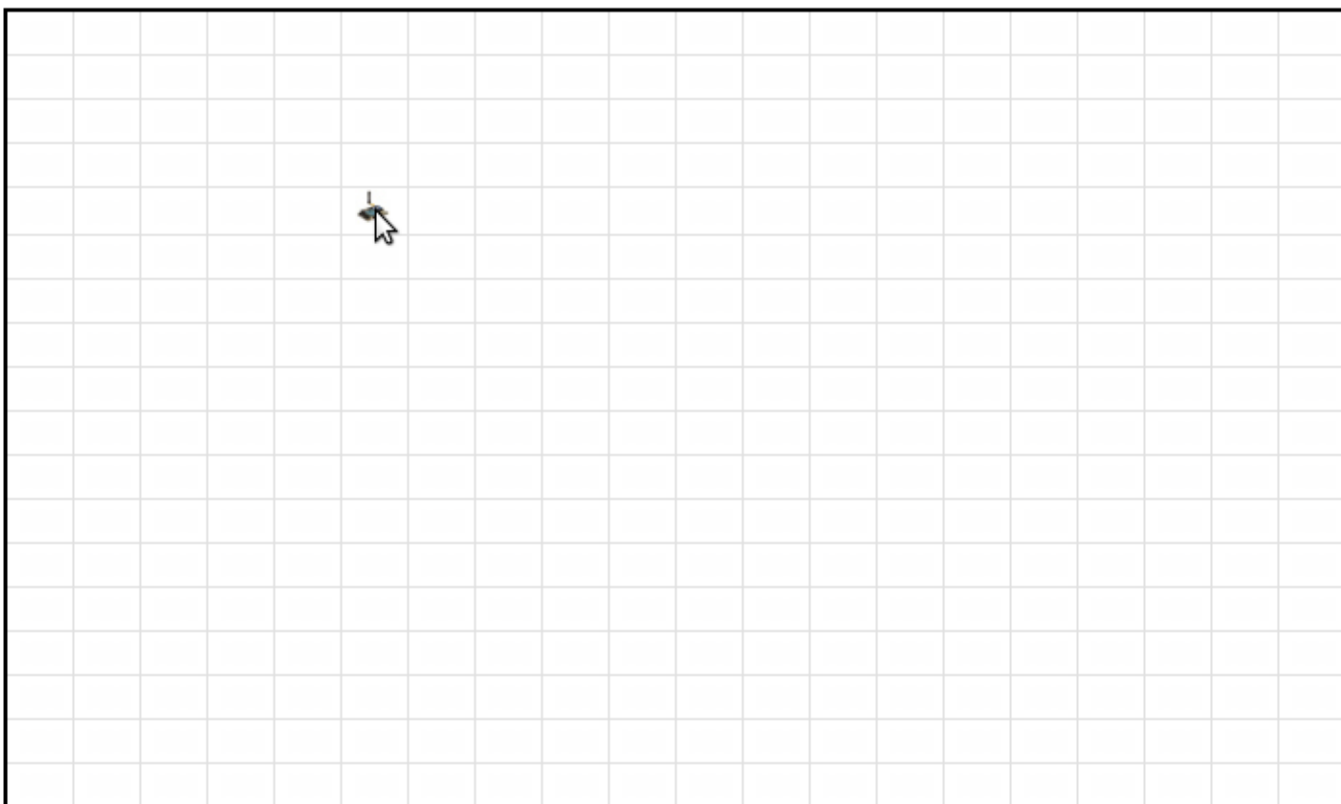
Después de hacer click en la zona seleccionada.



Haz click en la zona de la finca donde tengas el dispositivo waspmote. De esta manera te aparecera un dibujito del dispositivo.

De igual manera para quitar el dispositivo del mapa de la finca haz click y el dispositivo desaparecerá.

Red User 1: Red que controla la finca del usuario 1



Análogamente para quitar el dispositivo basta con hacer click en el dispositivo. Y con eso desaparecerá.

La última parte de este Manual de usuario está dedicada a la **Configuración** de los datos personales del usuario.

Aparecerá un formulario web relleno con los datos personales del usuario. Cuando el usuario modifique (o no) algún dato y le de a Aceptar el sistema se redirecciona al Muro.

Interfaz:



Datos personales

Nombre:

Apellidos:

Usuario:

Contraseña:

Rep. Contraseña:

Foto:

Telefono:

